

Vysoká škola Báňská – Technická univerzita Ostrava

Fakulta strojní

Katedra automatizační techniky a řízení

Využití PLC a PIC při realizaci modelu intelligentního rodinného domu

Use of PLC and PIC for Realization of Intelligent Family House

Student:

Bc. Miroslav Pawlenka

Vedoucí diplomové práce:

Ing. Jaromír Škuta, Ph.D.

Ostrava 2014

Zadání diplomové práce

Student: **Bc. Miroslav Pawlenka**
Studijní program: N2301 Strojní inženýrství
Studijní obor: 3902T004 Automatické řízení a inženýrská informatika
Téma: **Využití PLC a PIC při realizaci modelu inteligentního rodinného domu**
Use of PLC and PIC for Realization of Intelligent Family House

Zásady pro vypracování:

1. Navrhněte model inteligentního domu, definujte funkce tohoto systému, způsob jejich simulace a vyhodnocení.
2. Vyberte vhodné technické prostředky pro realizaci úlohy, seznamte se s prostředím pro jejich obsluhu a programování a s možnostmi jejich komunikace.
3. Navrhněte logickou strukturu celé úlohy, zohledněte možnost ovládání modelu standardními (běžně dostupnými) komunikačními prostředky.
4. Danou úlohu realizujte v podmínkách laboratoře katedry.
5. Zhodnoťte dosažené výsledky a navrhněte směr dalšího řešení.

Seznam doporučené odborné literatury:

BALÁTEĚ, J. *Automatické řízení*. Praha: Nakladatelství BEN, 2003, 654 s. ISBN 80-7300-020-2.
HERMANN, M., HANSEMAN, T., HUBNER, Ch. *Automatizované systémy budov : Sdělovací systémy KNX/EIB, LON a BACnet*. 1. Praha : Grada PUBLISHING, 2008, 264 S. ISBN 978-80-247-2367-9.
TYWONIAK, Jan. *Nízkoenergetické domy*. Praha : Grada, 2005, 200 s. ISBN 80-1101-X.
VLACH, J. *Řízení a vizualizace technologických procesů*. Praha: BEN, 1999, 160 S. ISBN 80-86056-66-X.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Jaromír Škuta, Ph.D.**

Datum zadání: 13.12.2013

Datum odevzdání: 19.05.2014



doc. Ing. Renata Wagnerová, Ph.D.
vedoucí katedry



doc. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě: 19. května 2014

.....

podpis studenta

Prohlašuji, že

- jsem byl seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- беру на ве́домі, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě diplomovou práci užít (§ 35 odst. 3).
- souhlasím s tím, že diplomová práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- было́ сје́днано, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona.
- было́ сје́днано, že užít své dílo –diplomovou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на ве́домі, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě: 19. května 2014



podpis

Bc. Miroslav Pawlenka

Okružní 1755/2

748 01 Hlučín

Anotace

PAWLENKA, M. *Využití PLC a PIC při realizaci modelu inteligentního rodinného domu: diplomová práce*. Ostrava: VŠB - technická univerzita Ostrava, Fakulta strojní, Katedra automatizační techniky a řízení, 2014, Vedoucí práce: Škuta, J.

Tato práce se zabývá řízením a monitorováním modelu inteligentního domu pomocí PLC firmy Siemens a jednočipového mikroprocesoru PIC. Cílem práce je návrh modelu inteligentního domu a způsobu jeho simulace pomocí mikroelektronických a mechanických prvků. Byly realizovány obvody pro simulaci regulace teploty, ovládání stejnosměrného motorku simulující otevírání brány, garážových vrat nebo stahující rolety na základě intenzity osvětlení, ovládání osvětlení a v neposlední řadě také moduly pro simulaci zabezpečujícího systému. Byla vytvořena aplikace ve SCADA systému Control Web 6, který komunikuje s PIC pomocí sériové komunikace a s PLC pomocí OPC komunikace. Dále byla vytvořena webová aplikace, která byla optimalizována pro prohlížení v internetových prohlížečích v tabletech a mobilních telefonech.

Práce obsahuje popis vývojového prostředí STEP7 pro PLC firmy SIEMENS. Jsou zde popsány programovací jazyky a editory, které je možno při vývoji aplikace použít, jednotlivé typy PLC firmy Siemens a základní údaje o jednočipových mikroprocesorech a jejich programování. Také jsou zde zmíněny základní rysy SCADA/HMI systému Control Web 6. V neposlední řadě jsou popsány jednotlivé desky plošných spojů, které byly v rámci návrhu modelu inteligentního domu realizovány, způsob jejich simulace a využití v rámci vytvořené aplikace v Control Webu 6.

Annotation

PAWLENKA, M. *Use of PLC and PIC for Realization of Intelligent Family House: Diploma Thesis*. Ostrava: VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation, 2014, Thesis head: Škuta, J.

The thesis deals with usage of PLC and PIC for realization of intelligent family house model. The aim was to design intelligent family house model with help of microelectronic components. In this thesis were made for example modules serving for temperature control, DC motor control, lights control or modules for security system. Data transfer between PIC and PC is realized by serial communication and between PLC and PC by OPC communication. Everything is controlled and monitored from SCADA/HMI system called Control Web 6. Application is accessible via Internet browser in local network. The view of application is adapted to viewing through mobile phones or tablets.

This thesis contains description of STEP7 development environment for Siemens PLCs, programming languages and editors which can be used for application development, Simatic controllers, peripheral interface controllers (PIC) and their programming and SCADA/HMI system Control Web 6. Last but not least there is a description of each created printed circuit boards (PCBs) and the way of their simulation within application created in Control Web 6.

Obsah

1	Úvod.....	11
2	Programovatelné automaty.....	12
2.1	Programovatelné logické automaty Siemens.....	14
2.2	Řídicí systém SIMATIC S7-300	14
2.3	Komunikační možnosti.....	15
2.3.1	Sběrnice AS-interface.....	16
2.3.2	Sběrnice Profibus.....	16
3	Programování ve vývojovém prostředí Siemens.....	19
3.1	Programy prováděné procesorem	20
3.2	Bloky uživatelského programu	21
4	Jednočipové mikroprocesory PIC	22
5	Control Web 6	26
5.1	Vývojové prostředí	27
5.2	Grafický editor.....	27
5.3	Datové inspektory.....	28
6	Inteligentní dům	29
6.1	Popis inteligentních domů	29
6.2	Návrh modelu inteligentní domácnosti.....	32
7	Realizace dílčích částí modelu	35
7.1	Napájecí modul.....	38
7.2	Rozhraní jednočipového mikroprocesoru.....	39
7.3	Snímač teploty TMP36GT9Z	40
7.4	Modul pro úpravu signálu ze snímače	40
7.5	Modul pro ovládání žárovky.....	41
7.6	Moduly pro ovládání DC motorku	42
7.7	Moduly pro ovládání osvětlení	43
7.8	Modul pro unifikaci napětí z LED diod.....	44

7.9	Snímač osvětlení	45
7.10	Napájení a indikace magnetických spínačů	45
7.11	Napájení a indikace snímače pohybu	46
7.12	Napájení a indikace světelných závor.....	46
8	Programování PIC	47
8.1	Čtení analogové hodnoty	48
8.2	Sériová komunikace se systémem Control Web 6	49
8.3	Algoritmy ovládání motorku a regulace teploty	52
9	Konfigurace a programování PLC	53
9.1	Program v PLC	53
9.2	Komunikace mezi PLC a Control Web 6	54
10	Návrh aplikace v prostředí Control Web 6.....	58
10.1	Vzhled a funkce vytvořené aplikace.....	59
10.2	Webová aplikace.....	64
10.3	Časování a popis programu	66
11	Závěr.....	68
12	Summary.....	70
13	Seznam použité literatury	72

Seznam použitých symbolů a zkratek

AI	Spojité vstup (<i>Analog input</i>)
ADC	A/D převodník (<i>Analog Digital Converter</i>)
AO	Spojité výstup (<i>Analog output</i>)
AS-interface	Průmyslový komunikační standard (<i>Actuator Sensor Interface, AS-i</i>)
BACNet	Komunikační protokol (<i>Building Automation and Control Networks</i>)
CPU	Centrální procesorová jednotka (<i>Central processor unit</i>)
CSS	Kaskádové styly (<i>Cascading Style Sheet</i>)
DC	Stejnoseměrný proud (<i>Direct Current</i>)
DHTML	<i>Dynamic HyperText Markup Language</i>
DI	Diskrétní vstup (<i>Discrete input</i>)
DO	Diskrétní výstup (<i>Discrete output</i>)
EEPROM	<i>Electrically erasable programmable read-only memory</i>
EIB	<i>European Installation Bus</i>
GSM	Globální systém pro mobilní komunikaci (<i>Global System for Mobile Communications</i>)
HMI	Rozhraní člověk – stroj (<i>Human Machine Interface</i>)
I2C	Sériová sběrnice (<i>Inter-Integrated Circuit</i>)
I/O	Vstupně / výstupní (<i>Input / Output</i>)
LED	Světlo emitující dioda (<i>Light Emitting Diode</i>)
LON	<i>Local Operating Network</i>
MDA	Kapesní mobilní zařízení (<i>Mobile Digital Assistant</i>)
MPI	<i>Multi – point interface</i>
OPC	Komunikační protokol (<i>OLE for Process Control</i>)
PCB	Deska plošných spojů (<i>Printed circuit board</i>)
PC	Osobní počítač (<i>Personal Computer</i>)
PDA	Kapesní mobilní zařízení (<i>Personal Digital Assistant</i>)

PIC	Jednočipový mikroprocesor (<i>Peripheral Interface Controller</i>)
PID	Proporcionálně – integračně – derivační
PIR	Pasivní infračervený senzor pohybu (<i>Passive infrared sensor</i>)
PLC	Programovatelný logický automat (<i>Programmable Logic Controller</i>)
PPI	<i>Parallel Peripheral Interface</i>
PROFIBUS DP	Průmyslová sběrnice (<i>Process Field Bus Decentralized Periphery</i>)
PWM	Pulsně šířková modulace (<i>Pulse Width Modulation</i>)
RAM	Paměť s přímým přístupem (<i>Random access memory</i>)
RISC	Redukovaná instrukční sada
ROM	Paměť jen pro čtení (<i>Read only memory</i>)
RS485	Standard sériové komunikace
SCADA	Dispečerské řízení a sběr dat (<i>Supervisory Control And Data Acquisition</i>)
SNMP	<i>Simple Network Management Protocol</i>
SPI	<i>Serial Peripheral Interface Bus</i>
TCP/IP	<i>Transmission Control Protocol/Internet Protokol</i>
USART	<i>Universal asynchronous receiver/transmitter</i>
USB	<i>Universal serial bus</i>
WWW	<i>World Wide Web</i>

1 Úvod

V dnešní době se s automatickým řízením můžeme setkat prakticky všude, kam se člověk podívá. Nemusí to být jenom součástí všech různých průmyslových odvětví, ale mohou to být například aplikace v domácnostech nebo celé inteligentní domy. Automatické řízení usnadňuje mnoha malým i velkým firmám práci zejména u složitých výrobních procesů. Bez automatického řízení by tedy měli práci velice ztíženou. Co se týče inteligentních budov a domů, slouží automatizace především pro snížení spotřeby a pro poskytování komfortu.

Další nedílnou součástí technologického procesu je jeho vizualizace. Ta se snaží poskytovat uživatelům nebo operátorům co nejefektivnější zpětnou vazbu mezi řídicím stanovištěm a řízeným procesem. Pomocí vizualizačních aplikací lze procesy nejen monitorovat, ale i řídit.

V této práci se nejdříve seznámíme s programovatelnými logickými automaty firmy Siemens. Bude provedeno rozdělení těchto PLC, bude zde popsáno vývojové prostředí Step 7 pro vytváření aplikací pro tyto PLC, bude popsána základní struktura programu a možnosti komunikace. Druhým použitým řídicím systémem bude jednočipový mikroprocesor řady PIC. Proto se s těmito mikroprocesory taktéž seznámíme. Následně bude zmíněn také systémem Control Web 6, jehož pomocí bude vytvořená aplikace monitorována a ovládána. V následující fázi budou navrženy některé úlohy simulující části inteligentního domu, navržena softwarová a hardwarová struktura a popsána realizace dílčích částí modelu. Nakonec zde bude ukázka vzhledu vytvořené aplikace v Control Webu 6 a charakterizována její funkčnost.

2 Programovatelné automaty

Následující kapitoly budou pojednávat o programovatelných logických automatech. Nejprve budou popsány základní hardwarové možnosti PLC obecně. Pak budou popsány jednotlivé PLC firmy Siemens.

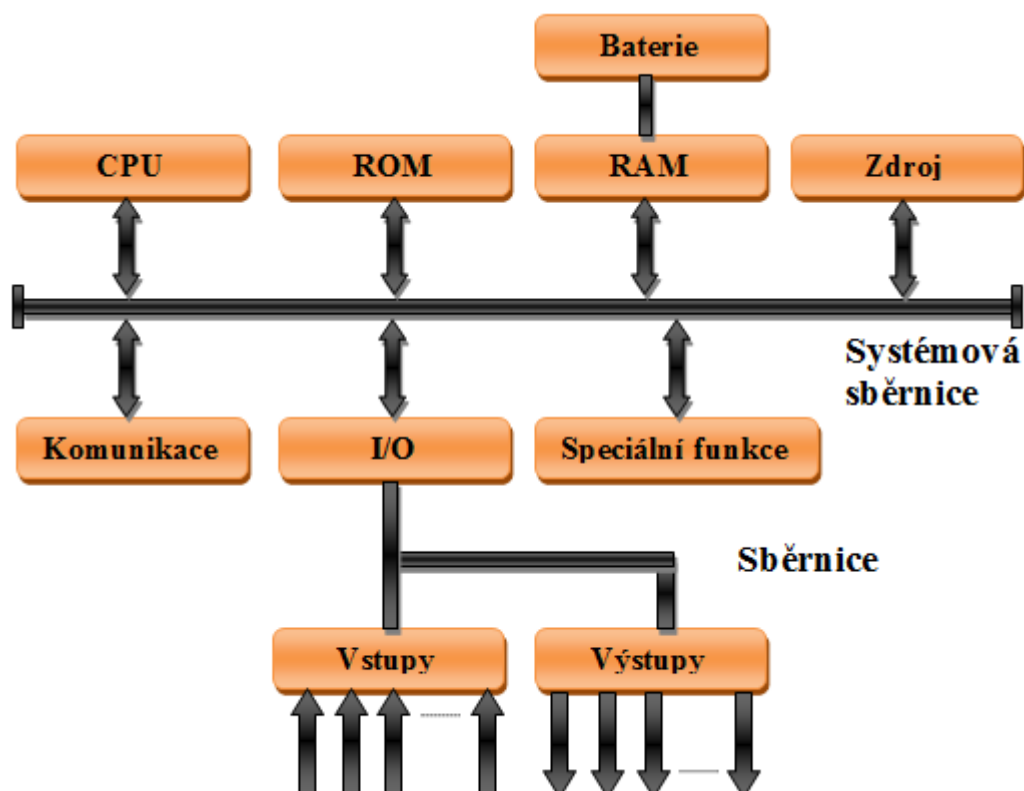
PLC, neboli programovatelné řídicí systémy, jsou v dnešní době jedny z nejpoužívanějších prostředků, které se využívají pro řízení nejen technologických procesů. Tyto řídicí systémy mají řadu výhod. Mají dobrý poměr cena/výkon, díky přídatným modulům lze nastavit široké možnosti komunikace a hlavně jsou vysoce robustní a spolehlivé v náročných podmínkách. Obecně se využívají pro logické řízení, regulaci, spojitě řízení, uchovávání a poskytování dat vyšším úrovním řízení atd.

PLC se dělí na kompaktní a modulární:

- **Kompaktní PLC** je takový systém, který v jednom modulu obsahuje CPU, digitální a analogové vstupy/výstupy a základní podporu komunikace. Rozšiřitelnost kompaktních PLC je omezena. Využívá se pro úlohy malého rozsahu.
- **Modulární PLC** je takový systém, kde jsou jednotlivé komponenty celku rozděleny do modulů. Celý systém PLC se potom skládá z modulů (napájecí zdroj, procesorová jednotka, moduly vstupů a výstupů atd.). PLC lze lehce rozšiřovat o přídatné moduly. Využívá se pro úlohy středního a velkého rozsahu.

Každé PLC se skládá z těchto částí, viz [KOZIOREK, J., CHROMČÁK L. 2007]:

- Centrální procesorová jednotka CPU
- Systémová paměť
- Uživatelská paměť
- Vstupní a výstupní jednotky
- Komunikační jednotky
- Speciální funkce
- Zdroj



Obrázek 1 - Bloková struktura PLC

PLC se dále dělí podle velikosti, viz [KOZIOREK, J., CHROMČÁK L. 2007]:

- **Mikro PLC** – nejmenší a nejlevnější kompaktní PLC systémy mají pevnou sestavu vstupů a výstupů, obvykle jen binárních. Mají kompaktní provedení, malé rozměry a nízkou cenu. Typickým použitím programovatelných automatů mikro PLC je realizace logické výbavy jednoduchých strojů a mechanismů, která se tradičně řešila pevnou reléovou logikou.
- **Malé PLC** – jsou to programovatelné automaty, které mají možnost zpracovávat desítky vstupů a výstupů. Jsou vhodné pro úlohy malého rozsahu a poskytují většinou již kompletní řadu funkcí pro logické i aritmetické operace. Konstrukčně se může jednat jak o kompaktní, tak o modulární PLC.
- **Střední PLC** – programovatelné automaty této kategorie jsou schopny zpracovávat stovky vstupně-výstupních signálů. Jsou vhodné pro řídicí úlohy středního rozsahu. Bývají většinou modulární konstrukce.
- **Velké PLC** – nejvyšší třída PLC, jsou schopny zpracovávat tisíce vstupů a výstupů. Používají se pro nejnáročnější aplikace. Bývají prakticky výhradně modulární konstrukce.

2.1 Programovatelné logické automaty Siemens

Řídicí systémy firmy Siemens nachází své uplatnění ve všech průmyslových odvětvích. Uplatňují se od jednoduchých průmyslových aplikací s jednoduchou logikou až po velmi rozsáhlé a složité technologické procesy, kde je třeba řídit a regulovat jak diskrétní, tak i spojité systémy. Následující kapitoly budou pojednávat o rozdělení řídicích systémů firmy Siemens.

Řídicí systémy firmy Siemens lze rozdělit do několika skupin podle toho, jaký proces se má řídit, viz [SIEMENS 2012]:

- Mikrosystémy
- Průmyslové automatizační systémy Simatic
- Řídicí systém Simotion
- Řídicí systémy Sinumerik
- Průmyslová PC a řízení na bázi PC
- Dále jsou dostupné systémy pro ovládání a vizualizaci

Řídicí systém Simotion je určen pro řízení pohonů a Sinumerik systémy jsou určeny pro řízení obráběcích strojů, krokových motorů a analogových pohonů.

2.2 Řídicí systém SIMATIC S7-300

Je to nejrozšířenější kompaktní nebo modulární řídicí systém firmy Siemens. Používá se hlavně pro automatizační úlohy středního rozsahu. Jádrem řídicího systému řady S7-300 je jednotka CPU, která zpracovává uživatelský program. Dle různých požadavků aplikace si uživatel může vybrat z následujících typů CPU, viz [SIEMENS 2012]:

- **Standardní CPU** - jsou osazeny programovacím a komunikačním rozhraním MPI nebo také rozhraním PROFIBUS. V novějších se také objevuje standard Ethernet, přes který lze také PLC programovat.
- **Kompaktní CPU** – tyto jednotky jsou osazeny digitálními a analogovými vstupy a výstupy a dále základními technologickými funkcemi jako jsou vysokorychlostní čítače, měření frekvence, polohování a PID regulace. Jako komunikační rozhraní se využívá PROFIBUS, MPI nebo RS485.
- **Bezpečnostní CPU (F-systémy)** – používají všude tam, kde je třeba zajistit co nejvyšší stupeň bezpečnosti obsluhy, výrobního zařízení či okolního prostředí

– např. je-li potřeba předejít nehodám a poškození zdraví či životního prostředí v důsledku poruchy. Využívá se zde bezpečnostní komunikace. Není nutná samostatná bezpečnostní komunikační linka. Program se vytváří standardně, ale je nutné dodržet normu IEC 61131-3 a speciální F – knihovnu.

- **Technologické CPU** – v tomto CPU jsou zahrnuty výkonné technologické funkce a funkce pro řízení pohybu a polohy. Umožňuje realizaci dynamického řízení pohybu ve více osách najednou. Obsahuje předprogramované funkce pro řízení pohybu, integrované vstupy a výstupy, komunikační rozhraní PROFIBUS-DP.

Programovatelný logický automat SIMATIC S7-300 lze také doplnit o rozšiřující moduly. Mohou to být například signálové moduly AI/AO a DI/DO, moduly rozhraní pro víceřadá uspořádání, funkční moduly pro zpracování časově náročných procesů nezávisle na CPU (rychlé čítání, měření frekvence, polohování, PID algoritmy), komunikační procesory pro propojení s dalšími přístroji sériovou linkou nebo propojení různých sběrníkových systémů.



Obrázek 2 - PLC SIMATIC S7-300, viz [SIEMENS 2013]

2.3 Komunikační možnosti

Komunikace mezi jednotlivými zařízeními řídicího systému je jeho základním stavebním kamenem. Požadavky jsou kladeny zejména na komunikaci s různými senzory, pohony, vzdálenými jednotkami vstupů a výstupů, SCADA systémy atd. V následujících odstavcích budou popsány základní komunikační rozhraní, které PLC firmy Siemens využívají, a to jsou AS-interface a PROFIBUS. Komunikační systémy lze rozdělit na několik úrovní, viz [KOZIOREK, J., CHROMČÁK L. 2007]:

- Actuator/Senzor Level – úroveň snímačů a akčních členů. Používají se zde sběrnice pro přenos binárních signálů. Klade se důraz na spolehlivost, jednoduchost a nízké náklady. Příkladem je AS-interface (Actuator Sensor Interface).
- Field Level – „polní“ úroveň. Komunikace mezi komponentami řídicím systémem v rámci jedné technologické sběrnice, vhodné pro přímé řízení, real-time komunikační systémy. Příkladem je Profibus, Interbus apod.
- Cell Level – komunikace v nebo mezi technologickými buňkami. Používají se zde výkonné komunikační sběrnice, umožňující rychlý přenos velkých objemů dat. Komunikace mezi PLC, PLC-HMI, HMI-vyšší úroveň apod. Nejběžnějším příkladem je Ethernet a jeho modifikace.

2.3.1 Sběrnice AS-interface

AS-interface je standard pro komunikaci mezi akčními členy a snímači na první úrovni řízení strojů, výrobních linek a technologických procesů, jenž je určen především pro aplikace s převahou binárních signálů a logického řízení, viz [AUTOMA 2004].

Přenos dat probíhá po jednom společném vodiči, kterým je nestíněný nekroucený dvoužilový kabel s izolací zvláštním způsobem profilovanou, umožňující snadnou, spolehlivou a rychlou montáž a přenos signálů i napájení. Maximální délka je 100m. Komunikační síť je typu Master – Slave. Na jedné stanici Master může být až 31 (62 pro verzi 2.1) stanic podřízených – Slave. Zprávy se přenášejí na základě cyklického vyzývání s tím, že doba trvání cyklu nesmí překročit 5ms (10ms u verze 2.1). Mezi výhody patří:

- Minimální náklady na instalaci a provoz
- Rychlá a bezpečná instalace
- Flexibilní konfigurace (široké spektrum komponent, vysoká modularita)
- Integrace bezpečnostních technologií

2.3.2 Sběrnice Profibus

Průmyslová sběrnice Profibus je v současnosti jednou z nejrozšířenějších komunikačních standardů a je určena především pro všechny oblasti automatizace, což může být řízení různých technologií a procesů popřípadě automatizovaných budov a inteligentních domů. Vychází z komunikačního modelu ISO/OSI. Z důvodu časové

optimalizace definuje z tohoto modelu pouze vrstvy fyzickou, linkovou a aplikační, viz [KOZIOREK, J., CHROMČÁK L. 2007]:

- Fyzická vrstva definuje fyzické spojení mezi zařízeními včetně mechanických a elektrických vlastností tohoto spojení a současně je v této vrstvě definována topologie sítě. Profibus podporuje přenos po sběrnici RS-485, optickém vláknu a pro výbušné prostředí po proudové smyčce IEC 1158-2.
- Linková vrstva (Fieldbus Data Link) definuje mechanismus přístupu účastníka na přenosové médium (token passing, master-slave) a zabezpečuje tvorbu zprávy na úrovni bitového řetězce včetně generování kontrolních částí.
- Aplikační vrstva je nejvyšší vrstvou v referenčním modelu ISO/OSI. Poskytuje jednotlivé služby, nezbytné pro realizaci komunikace z hlediska uživatele.

Průmyslová sběrnice Profibus existuje ve třech variantách, viz [KLEČKA R. 2004]:

- **Profibus DP (Decentralized Periphery)**

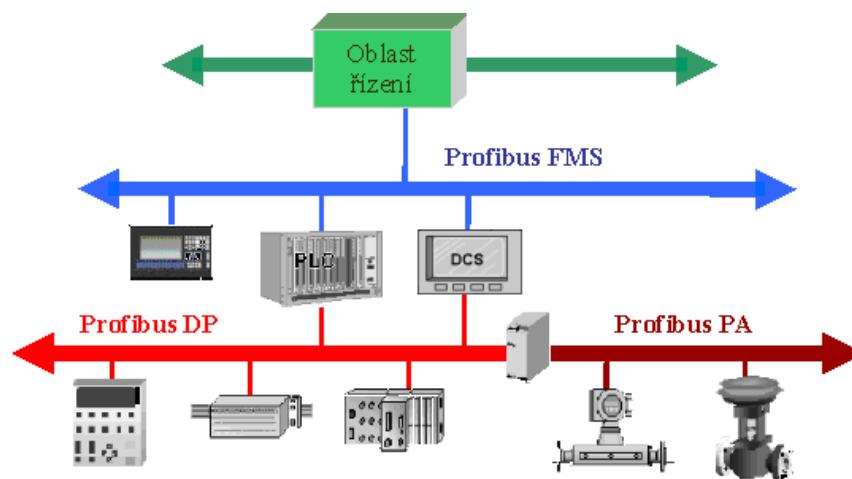
Nejjednodušší a nejrozšířenější variantu sběrnice Profibus, která se používá jako sensor bus. Tato varianta je určena pro rychlou komunikaci typu master-slave a je vhodná zejména pro rychlý přenos signálů z procesu pomocí decentralizovaných periférií a odloučených I/O jednotek. Komunikačním médiem je buď kroucená dvojlinka (standard RS-485) nebo optické vlákno při rychlosti až 12 Mb/s.

- **Profibus FMS (Fieldbus Message System)**

Patří do skupiny průmyslových sběrnic typu Fieldbus. Umožňuje práci s velkou množinou služeb pro manipulaci s daty, programy a alarmy. Komunikačním médiem je buď kroucená dvojlinka (standard RS-485), nebo optické vlákno, avšak rychlost je už nižší.

- **Profibus PA (Process Automation)**

Patří do skupiny průmyslových sběrnic typu Fieldbus. Používá rozšířenou normu Profibus DP a je určena pro řízení pomalých procesů zvláště ve výbušném prostředí. Aby bylo možné síť využít také v prostředí s nebezpečím výbuchu, je použita i speciální fyzická vrstva - proudová smyčka.

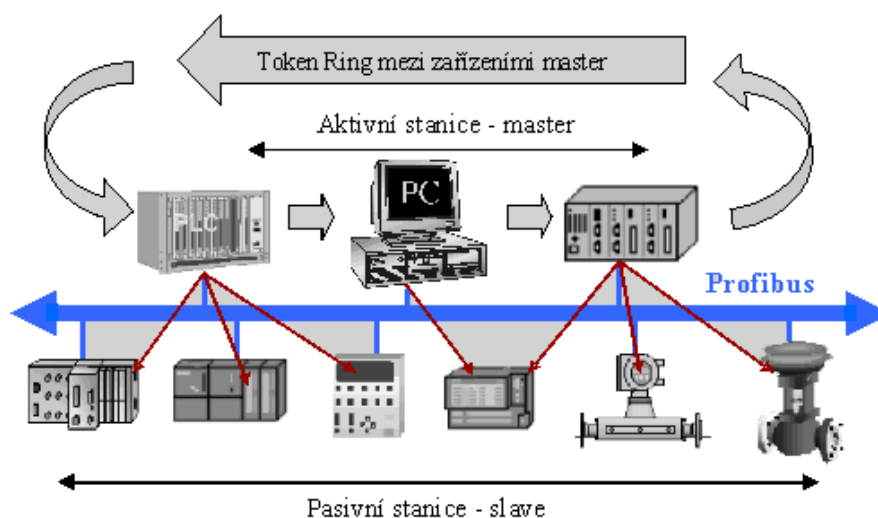


Obrázek 3 - Využití sběrnice Profibus, viz [KLEČKA R. 2004]

Stejně jako v počítačových sítích, i zde musí být zabezpečena kolize dat, kdy začne současně vysílat více účastníků najednou. V danou chvíli má tedy právo vysílat pouze jedno zařízení, které by mělo mít na dokončení komunikace dostatečně dlouhou dobu. Tato doba by měla být co nejrychlejší. Základními řešeními jsou metody řízení přístupu na sběrnici, viz [KLEČKA R. 2004]:

Metoda token passing – (předávání pověření v logickém kruhu) pro komunikaci mezi aktivními zařízeními

Metoda master-slave – (centrálně řízené dotazování) pro komunikaci mezi aktivním a jemu přidělenými zařízeními



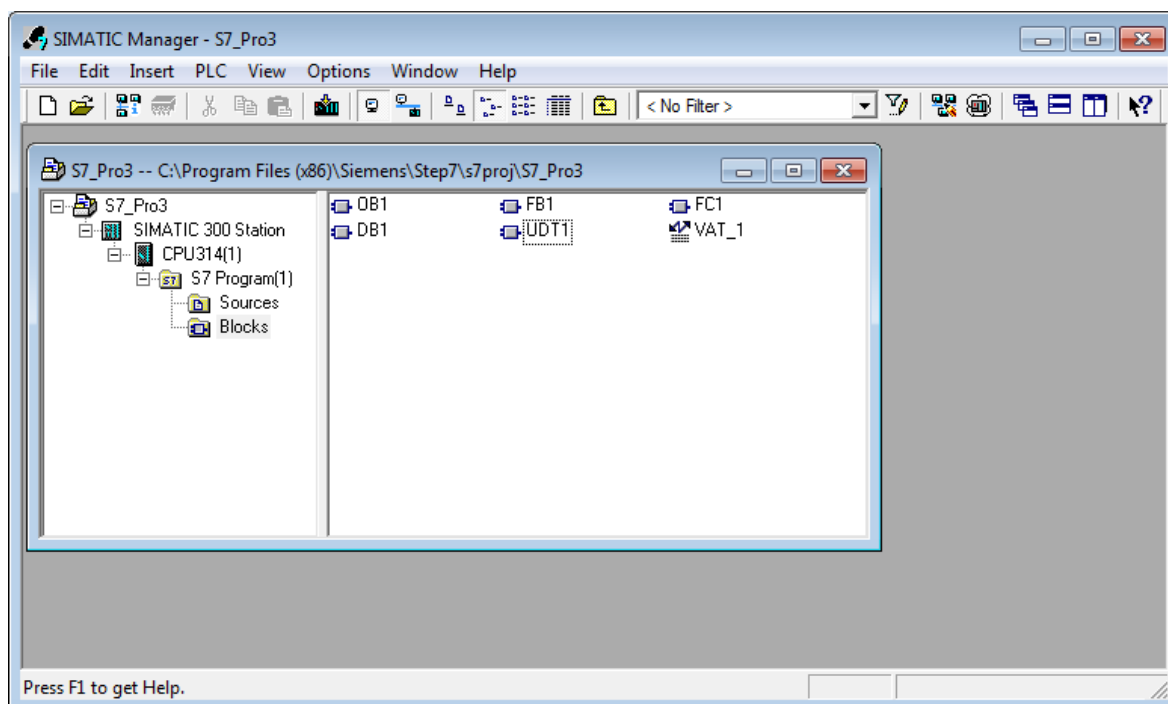
Obrázek 4 - Princip přístupu sběrnice Profibus, viz [KLEČKA R. 2004]

3 Programování ve vývojovém prostředí Siemens

Vývojové prostředí STEP 7 je určeno pro vytváření aplikací a správu řídicích systémů SIMATIC S7-300/400. STEP 7 je složen z těchto hlavních částí, viz [SIEMENS 2009]:

- **SIMATIC Manager** – Slouží pro správu všech nástrojů spadajících pod celý projekt.
- **Symbol editor** – Editor pro správu globálních proměnných. Umožňuje definovat symbolické označení a komentáře pro zpracovávané signály.
- **Hardware configuration** – Editor pro nastavení hardwarové konfigurace a její následné diagnostiky.
- **NetPro (Network Configuration)** – Editor pro nastavení komunikace přes MPI, PROFIBUS nebo PROFINET.

Vývojové prostředí obsahuje výkonné nástroje a funkce pro veškeré etapy procesu vývoje, jakými jsou konfigurace hardwaru, nastavování komunikace, programování, testování, zprovoznění celého projektu, archivování, provozní a diagnostické funkce. Velkou výhodou programování je dostupnost pěti různých programovacích jazyků, které se dají kombinovat. Na každou danou část programu je tedy vhodné použít nejvhodnější programovací jazyk, který lze vybrat dle normy IEC 61 131.



Obrázek 5 - SIMATIC Manager

Seznam programovacích jazyků:

- **STL – Statement List** – instrukční list (obdoba assembleru). Má nejlepší možnosti pro optimalizaci běhu programu, avšak oproti grafickému programování je složitější, jelikož musí programátor znát příkazy a registry. Je paměťově úsporný a zajišťuje rychlost zpracování. Využívá se pro tvorbu funkcí, které jsou opakovaně používány. Do tohoto programovacího jazyku lze přeložit všechny ostatní programovací jazyky.
- **FBD – Function Block Diagram** – funkční blokový diagram patří mezi grafické jazyky. Je vhodný pro realizaci logických řídicích sekcí programu a zpracování vstupních a výstupních signálů. Grafický zápis programu je přehledný. V SIMATIC Manageru jsou dostupné bloky pro bitovou logiku, komparátory, čítače, časovače, bloky realizující regulaci atd.
- **LAD – Ladder Diagram** – žebříčkový diagram (liniové nebo reléové schéma). Je to grafický programovací jazyk, velmi přehledný. Nevýhodou je programování složitější logiky, což má za následek nepřehlednost schématu. Jasně definuje posloupnost operací, a proto lze v online režimu snadno pozorovat, jak se program chová. Není vhodný pro aritmetické operace.
- **SCL – Structured Control Language** – Vyšší programovací jazyk vycházející z jazyka Pascal a odpovídající strukturovanému textu, vhodný spíše pro rozsáhlejší algoritmy s matematickými funkcemi nebo úlohy zpracování dat. Jednoduchá práce s textovými řetězci a ideální pro zpracování datové komunikace a zpracování analogových signálů.
- **GRAPH** – sekvenční funkční diagram. Je vhodný pro sekvenční aplikace, kdy vstup do další části programu závisí na vykonání předchozí částí a splnění zadaných podmínek. Pro složitější programy je nepřehledný.

3.1 Programy prováděné procesorem

V paměti PLC jsou vždy 2 hlavní části, a to operační systém a uživatelský program. Operační systém organizuje funkce a sekvence, které nesouvisí s úlohou řízení. Úlohy operačního systému jsou tyto, viz [SIEMENS: SIMATIC 2004]:

- Zajištění startu systému
- Aktualizování tabulky vstupů a poskytování tabulky výstupů
- Volání uživatelského programu

- Detekování přerušení a volání bloků s přerušením
- Detekování a řešení chyb
- Spravování místa v paměti
- Komunikace s přístroji a ostatními komunikačními rozhraními

Uživatelský program musí do procesoru nahrát programátor. Požadavky uživatelského programu zahrnují, viz [SIEMENS: SIMATIC 2004]:

- Specifikace počátečních podmínek procesoru (například počáteční signál s určitou hodnotou)
- Zpracování dat (například logická kombinace binárních signálů, načtení vyhodnocovacích analogových signálů, specifikace binárních signálů pro výstup, poskytování analogových hodnot)
- Specifikace reakcí na přerušení
- Vyřešení poruch za normálního běhu programu

3.2 Bloky uživatelského programu

Vývojový software STEP 7 umožňuje strukturovat program, jinak řečeno rozdělit program na jednotlivé sekce, obsahující vlastní programové sekce. Výhodou je, že jsou rozsáhlé programy srozumitelnější. Jednotlivé bloky mohou být použity vícekrát v programu i v jiných projektech. Dále mohou být jednodušeji modifikovány a jednoduše otestovány. Uživatelský program se může skládat z následujících bloků, viz [SIEMENS: SIMATIC 2004]:

Tabulka 1 - Bloky uživatelského programu

Blok	Základní popis funkce
Organizační bloky (OB)	Určují strukturu uživatelského programu. Řídí start systému, cyklické, časové provádění, obsluhují přerušení atd. Jsou volány OS.
Systémové funkční bloky (SFB) a systémové funkce (SFC)	Jsou integrovány v CPU a umožňují přistupovat k důležitým systémovým funkcím.

Blok	Základní popis funkce
Funkční bloky (FB)	Bloky s „pamětí“, které lze naprogramovat.
Funkce (FC)	Obsahují programové části, které jsou často používány. Oproti funkčním blokům (FB) nemají vlastní paměťovou oblast.
Stupňovité bloky dat (instance DB)	Jsou asociovány tehdy, když jsou volány systémové funkční bloky nebo funkční bloky. Vytvářejí se automaticky během kompilace.
Bloky dat (DB)	Oblasti pro ukládání dat. Dále pro data, která jsou přiřazena funkčním blokům. Sdílená data mohou být také definovány a použity jako bloky.

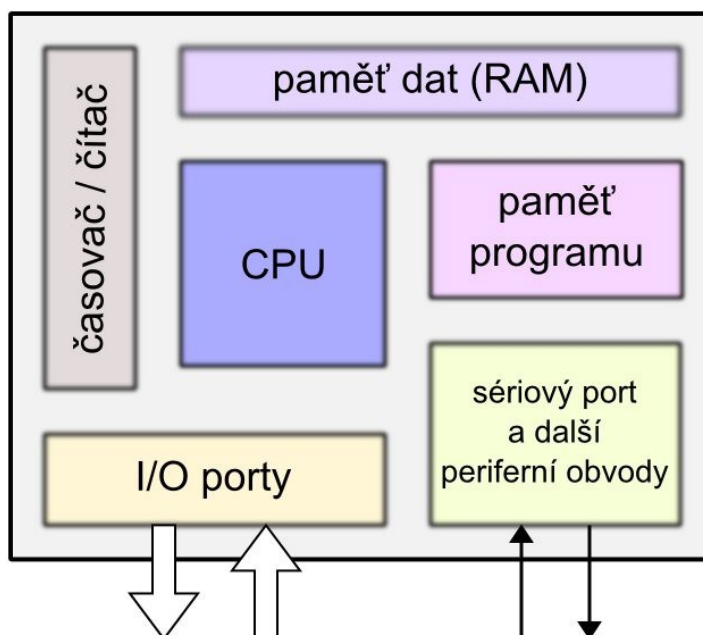
Nejdůležitějším organizačním blokem je blok OB1. Ten obsahuje cyklicky zpracováváný uživatelský program, který volá jednotlivé ostatní naprogramované bloky a funkce. Blok OB1 má nejnižší prioritu. Proto může být přerušeno vykonávání tohoto bloku voláním jiného organizačního bloku s vyšší prioritou. Po vykonání tohoto bloku se program vrací zpět do místa, kde byl přerušen. Obecně mohou všechny bloky s vyšší prioritou přerušit bloky s nižší prioritou.

4 Jednočipové mikroprocesory PIC

Jednočipový mikroprocesor je součástka v podobě jednoho integrovaného obvodu, která se dá programovat. Nespornou výhodou těchto mikropočítačů je cena, nízká spotřeba, snadná programovatelnost, pracovní rozsah ve velkém rozmezí teplot atd. Využívají se například v domácích spotřebičích, a to zejména díky své velikosti. Dále se používají v automobilech nebo měřicích a řídicích systémech. Řada mikroprocesorů PIC je založena na Harvardské architektuře (paměť programu a paměť dat jsou odděleny) a jsou vyráběny firmou Microchip Technology.

Základní vlastnosti, viz [ŠPRINGL, V. 2011]:

- Malé množství vysoce optimalizovaných instrukcí – redukovaná instrukční sada RISC
- Většina instrukcí je vykonána v jednom instrukčním cyklu
- Periferie a konfigurační registry jsou mapované do datové paměti mikrokontroléru
- Obsahují jeden pracovní registr W, který slouží pro provádění veškerých aritmetických operací



Obrázek 6 - Základní struktura mikrokontroléru, viz [ŠPRINGL, V. 2011]

Základní části jednočipových mikroprocesorů, viz [ŠPRINGL, V. 2011]:

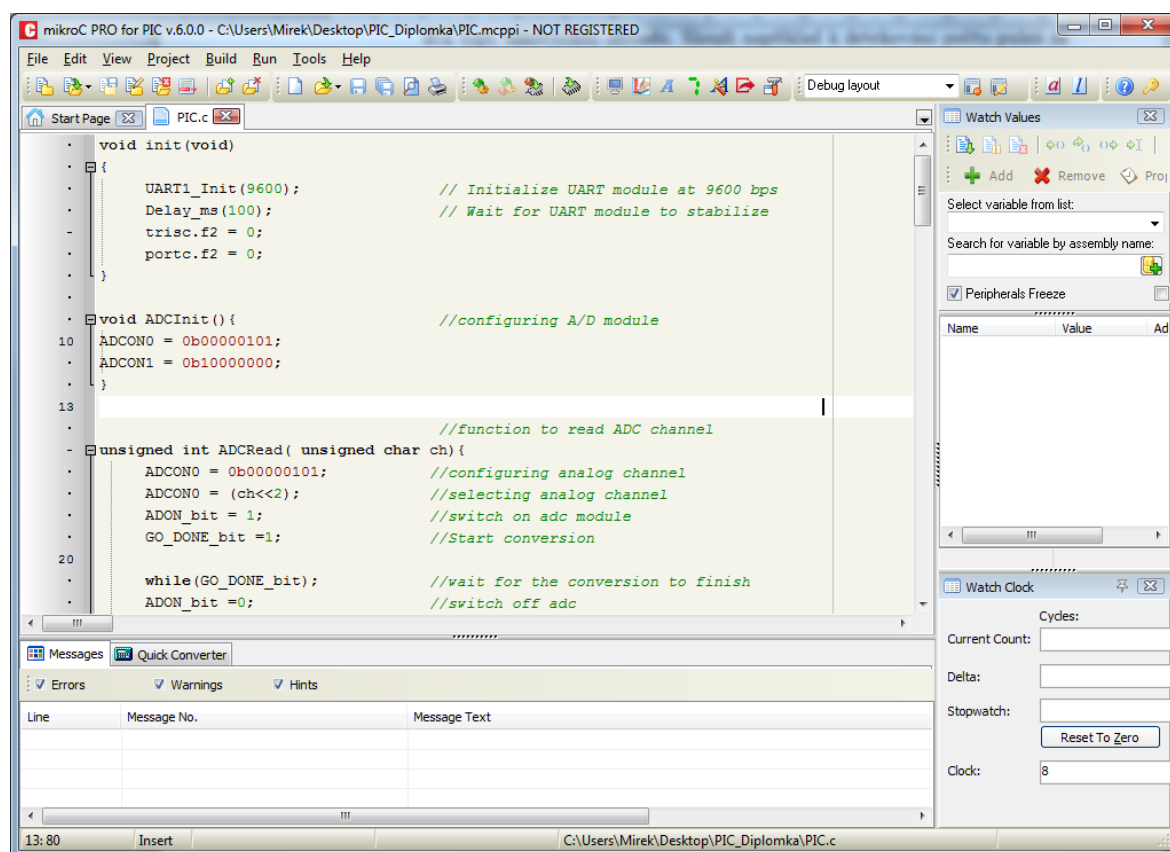
- **Procesor (CPU)** – nejdůležitější část mikrokontroléru. Řídí veškerou činnost počínaje načítáním, dekódováním a vykonáváním instrukcí až po čtení a zápis dat do paměti, komunikaci se vstupně/výstupními porty a dalšími periferními zařízeními.
- **Paměť programu a dat** – Slouží k uložení vlastního programu v podobě instrukcí, které řídí veškerou činnost, a k uložení dat, se kterými mikrokontrolér pracuje. Programová paměť je nejčastěji realizována jako paměť FLASH. Výhodou je, že ji lze v případě potřeby přeprogramovat. Pro datovou paměť k uložení výsledků aritmetických nebo logických operací a dalších dat se využívá paměť RAM. Nevýhodou je, že po vypnutí napájecího

napětí je tato paměť vymazána, a proto se u některých mikrokontrolérů využívá paměti EEPROM, která důležitá data uchovává i po odpojení od napájení.

- **Vstupně/výstupní rozhraní** – Toto rozhraní umožňuje mikrokontroléru komunikovat s externími zařízeními. Jednotlivé piny lze nastavit jako vstupní nebo výstupní
- **Obvody čítačů a časovačů** – Většinou mikrokontrolér obsahuje jen jeden nebo dva typy takovýchto obvodů. Slouží například k detekování počtu pulsů za daný časový interval, měření časového intervalu apod.
- Další komponenty

Jednočipové mikropočítače se dále mohou dělit na čtyřbitové, osmibitové, šestnáctibitové nebo třicetidvoubitové. Co se týče programování, tak je možné programovat více způsoby. Programovací jazyky se dělí na nízkoúrovňové a vysokoúrovňové. Mezi nízkoúrovňové programovací jazyky patří Assembler. Ten je tvořen sadou instrukcí, které jsou potřebné pro vytvoření strojového kódu. Výhodou je efektivní využívání paměti a vysoká rychlost provádění programu. Nevýhodou je závislost na dané platformě a složitý zápis. Mezi vysokoúrovňové jazyky se dají zařadit například Visual Basic, C, C++, C#, Pascal, Delphi, Java atd. Na rozdíl od nízkoúrovňových jazyků není kód závislý na hardwaru a je tedy snadno přenositelný. Přenositelnost zajišťuje kompilátor, který překládá uživatelem napsaný zdrojový kód do kódu strojového. Samotné programování je díky logické struktuře snazší.

Pro programování mikroprocesorů PIC existuje mnoho vývojových prostředí. Nejznámější jsou MPLAB IDE firmy Microchip nebo MikroC, MikroBasic a MikroPascal firmy Mikroelektronika. Pro programování mikroprocesoru v této práci byl použit software Mikroelektronika MikroC. Při založení projektu je důležité nastavit taktovací frekvenci a zvolit typ použitého PIC, a to z důvodu, že každý mikroprocesor má jiné funkce a jiné dostupné knihovny.



Obrázek 7 - Vývojové prostředí MikroC

Dalšími možnostmi vývojového prostředí jsou zobrazování speciálních funkčních registrů nebo sledování obsahu proměnných. Výhodným prostředkem pro simulaci komunikace pomocí sériové linky je USART Terminal, pomocí něhož lze sledovat příchozí komunikaci i odesílat na výstup znaky nebo řetězce.



Obrázek 8 - Programátor Presto, viz [ASIX. Presto - obrázky]

Po napsání programu bylo nutné jej nahrát do mikroprocesoru. Bylo využíváno programátoru Presto a programu Asix UP.

5 Control Web 6

Systém Control Web patří do skupiny SCADA/HMI systémů. Tento vývojový software vyvinula firma Moravské přístroje. Control Web je nástroj, který slouží nejen pro vizualizaci a sběr dat, ale umožňuje i vytváření aplikací pro přímé řízení strojů v reálném čase – vytváří rozhraní člověk – stroj, viz [MORAVSKÉ PŘÍSTROJE a.s. 2005].

Control Web je nezávislý na hardware. Díky patřičným ovladačům může komunikovat s jakýmkoliv zařízením, např. PLC, I/O moduly, měřicí karty, WWW servery a podobně, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010A].

Tak, jako ostatní SCADA/HMI systémy, má i Control Web možnost využívání tlustého i tenkého klienta, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010A].

Control Web Runtime – tlustý klient:

- Sdílená data po síti, volání vzdálených metod
- Zálohování, synchronizace dat
- Vzdálený přístup
- Tvorba aplikací client/server nebo peer-to-peer

Přístup přes WWW prohlížeč – tenký klient

- Zabudovaný http server v systému Control Web
- Serverové aplikace pro klienty na PC i mobilních telefonech
- Umožňuje použití HTML, DHTML/CSS, Java, Active-X, atd.

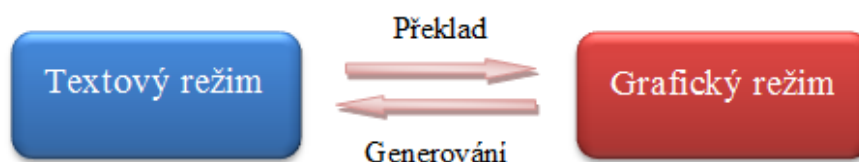
Systém Control Web dokáže provozovat modulární, distribuované a synchronizované aplikace, což je v mnoha případech výhodné. Následně je popsán význam těchto pojmů, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010B].

- **Modulární** aplikace se skládá z více modulů. Systém Control Web považuje za modul každý samostatný soubor. To znamená, že nejjednodušší aplikace je jednomodulární. Taková aplikace se celá nachází v jednom zdrojovém souboru.
- **Distribuovaná** je aplikace, ve které jednotlivé její sdílené součásti běží na různých počítačích. Distribuovaná aplikace v systému Control Web vždy běží v modelu klient-server.
- **Synchronizovaná** je distribuovaná aplikace, ve které jsou některé její sdílené součásti udržovány v synchronním stavu (obsahují v jednom okamžiku stejná data).

Nesynchronizovaná distribuovaná aplikace má přesně dáno, která součást musí být server a která klient. U synchronizované aplikace toto stanoveno není. Funkce jednotlivých distribuovaných součástí se může měnit z klienta na server nebo naopak. Vždy lze ale říci, která část je která.

5.1 Vývojové prostředí

Control Web 6 využívá dvoucestného programování. To umožňuje pracovat ve dvou vývojových prostředích - grafický editor a textový editor. Tyto dvě prostředí jsou navzájem propojeny, a tudíž změny, provedené v jednom prostředí, se okamžitě projeví ve druhém prostředí a opačně. Toto se nazývá překlápění. Aplikace tedy může existovat jak v grafické, tak v textové podobě. Při vytváření aplikace se pracuje především v grafickém editoru. Textový editor pak slouží hlavně jako doplnění grafického, čímž se dá vytvořit dokonalá komplexní aplikace. Avšak zdrojový tvar aplikace je textový. Jelikož můžeme při psaní zdrojového kódu napsat jakýkoliv text, tak musí existovat nástroj, který kontroluje, zda je text napsán správně a zda je přeložitelný. K tomuto účelu slouží překlad. Zajistí se tím tedy přechod z textového režimu na grafický. Opačný způsob, tedy přechod z grafického režimu na textový, se nazývá generování. Generování vytvoří z grafické (binární) podoby zdrojový text, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010B].



Obrázek 9 - Překlad a generování prostředí

5.2 Grafický editor

Grafický editor se skládá z několika částí. Tyto části spolu úzce souvisí a absence některé z nich by měla fatální dopad na aplikaci, která by se prakticky nedala vytvořit. Každá z těchto částí tedy hraje klíčovou roli. Komunikaci s reálným světem, použitá data, vnitřní údaje a nastavení parametrů aplikace shrnují *Datové inspektory*. Nabídku přístrojů spolu s jejich základním popisem (například procedur) obsahuje *Paleta přístrojů*. Prostor pro vizuální návrh a řízení běhu aplikace pokrývá hlavní část vývojového prostředí, vlastní *Grafický editor*. Každý objekt lze rychle změnit a upravit v *Inspektoru přístroje*, Grafický

editor si lze také představit jako plochu, do které se vkládají různé přístroje a podle potřeby lze měnit jejich vlastnosti, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010B].

5.3 Datové inspektory

Využívají se hlavně k vytváření všech datových elementů a proměnných, se kterými pak pracují jednotlivé přístroje v grafickém editoru. Absence datových elementů by zapříčinila, že by byl vzhled aplikace pouze statický. Další funkcí datových inspektorů je nastavení ovladačů pro komunikaci. Všechny datové proměnné vytvořené v datovém inspektoru jsou globální. Z toho plyne, že jsou přístupné všem použitým přístrojům nebo také jejich procedurám. Control Web rozeznává čtyři hlavní skupiny datových typů, viz[MORAVSKÉ PŘÍSTROJE a.s. 2010B].

- **logické hodnoty** - obsahují binární hodnotu, tedy hodnotu vyjadřující stav zapnuto/vypnuto (true / false). Datový typ - boolean.
- **číselné hodnoty** - dělí se podle druhu čísel na samostatné datové typy. Čísla jsou buď desetinná (shortreal, real), nebo celá (shortint, shortcard, integer, cardinal, longint, longcard), a to buď se znaménkem (shortreal, real, shortint, integer, longint) nebo bez znaménka (shortcard, cardinal, longcard).
- **řetězce znaků** - obsahují skupiny písmen bez omezení délky. Řetězce se zapisují do apostrofů (' ').
- **datový typ buffer** - obsahuje blok paměti, který je nějak vnitřně strukturován. Systému Control Web definuje pouze základní parametry (například délku bufferu), zbytek vyplývá z konkrétního použití.

6 Inteligentní dům

V této části budou popsány možnosti a jednotlivá řešení inteligentních domácností, co všechno je do takového domu zahrnuto, prostředky pro realizaci a příklad reálného systému, který se v praxi používá. Dále bude navržen model inteligentního domu.

6.1 Popis inteligentních domů

Inteligentní domy se v dnešní době začínají objevovat čím dál více. Takový dům neslouží jen pro vytvoření pohodlí uživatelů či bezpečnosti, ale také pro snížení nákladů spojených s provozem domácnosti.

Mezi základní aplikace inteligentních domácností patří vytápění, klimatizace, ventilace, ovládání žaluzií, osvětlení, zabezpečení domu, možnost vypínání zásuvek za účelem snížení spotřeby a mnoho dalších aplikací. Mezi zabezpečení patří požární čidla, kamerový systém, senzory pohybu v místnostech, magnetické snímače pro detekci otevření dveří nebo oken a další. Vše je pak možno ovládat skrze počítač, notebook, tablet, panel umístěný na stěně, mobilní telefon či dálkový ovladač. Ovládání skrze aplikaci v mobilním telefonu či tabletu je v dnešní době nejčastější. Stav jednotlivých aplikací lze sledovat samozřejmě i vzdáleně skrze Internet. Tomu všemu ale předchází návrh a integrace veškeré elektroinstalace a kabeláže.

Veškeré systémy automatizace budov využívají různých komunikačních protokolů. Nyní budou popsány některé nejvíce používané sběrnice, které se v praxi automatizace budov používají.

- **Sběrnice EIB**

Evropská instalační sběrnice EIB (European Installation Bus) vznikla z elektroinstalační sběrnice Instabus firmy Siemens. Sběrnice EIB má decentralizovanou strukturu s liniovou, kruhovou nebo větvenou topologií. K jedné větvi může být připojeno až 64 zařízení. Informace po sběrnici jsou předávány v tzv. telegramech (zprávách). Programování jednotlivých účastníků a celého systému EIB se provádí pomocí programu ETS (EIB Tool Software). Jako základní přenosové médium je použito krouceného páru vodičů (označováno jako EIB-TP). Dále může být použito síťové vedení (EIB-PL - Power Line) nebo přenos signálů rádiiem (EIB-RF - Radio Frequency). Výhodou sběrnice je ta, že mohou být bez problému propojovány zařízení různých výrobců, viz [MATZ, V. 2010].

- **Sběrnice KNX**

Základem pro tuto sběrnici byla sběrnice EIB. Všechny komponenty, které vyhovují EIB, vyhovují i sběrnici KNX. Proto se často sběrnice označuje KNX-EIB. Oproti EIB má KNX daleko více funkcí. Je zde možnost využití dalších přenosových médií nebo integrace různých zařízení, viz [MATZ, V. 2010].

- **Sběrnice LON**

Sběrnice LON je otevřený decentralizovaný sběrnice systém využívající sériového přenosu dat (zpráv). Skládá se z uzlů (řídící systémy, regulátory), které si mezi sebou vyměňují informace. Každý regulátor obsahuje univerzální čip, obsahující neuronový čip a připojení na sběrnici. Neuronový čip obsahuje tři osmibitové procesory, paměti, časovací jednotku, vstupní/výstupní část a komunikační sběrnici. V praxi se sběrnice LON s výhodou využívá v aplikacích, kde je kladen nárok na délku sběrnice (nikoliv na rychlost přenosu dat), viz [MATZ, V. 2010].

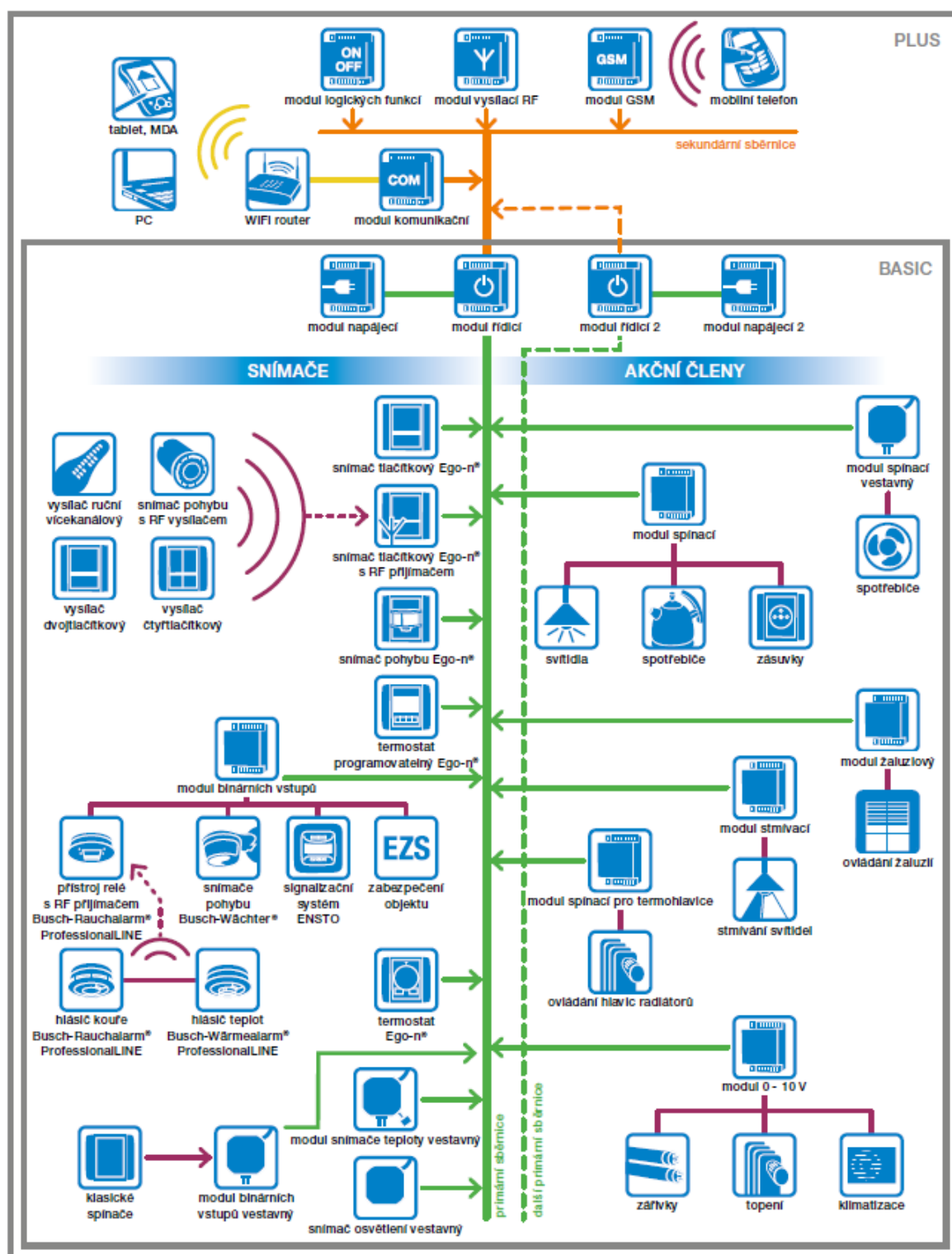
Sběrnice a různých dalších protokolů je pochopitelně více. Mezi dalšími známými komunikačními protokoly, které se často používají, jsou například BACNet, Modbus, OPC nebo SNMP.

Příkladem systému, který je vhodný do rodinných domů, bytů či kanceláří je systém firmy ABB jménem Ego-n. Tento systém umožňuje, viz [ABB 2011]:

- Řízení spínání a stmívání osvětlení
- Detekci vnitřního i venkovního pohybu
- Řízení pohonu žaluzií, předokenních rolet a markýz
- Řízení vytápění a chlazení, klimatizace
- Ovládání libovolných spotřebičů (s možností vzájemného blokování na základě zvolených priorit), logické, centrální a časové funkce
- Vizualizaci a dálkové ovládání (pomocí vhodného tabletu, PC, kapesního počítače PDA nebo MDA)
- vzdálený přístup a ovládání (prostřednictvím GSM a internetu)

Systém má dva typy sběrnice (primární a sekundární). Primární sběrnice je základní a je možné na ni připojit až 64 prvků systému. Na primární sběrnici jsou připojeny vstupy – snímače (např. tlačítkové, digitální) a výstupy – akční členy (např. stmívací nebo spínací moduly). Systém je tvořen dvěma úrovněmi – Basic a Plus v závislosti na rozsahu. Úroveň

Basic má jeden modul a lze jej programovat bez použití PC. U úrovně Plus je k programování nutný komunikační modul a připojení k PC, viz [ABB 2011].



Obrázek 10 - Základní struktura systému firmy ABB Ego-n, viz [ABB 2011]

6.2 Návrh modelu inteligentní domácnosti

Model inteligentního domu bude simulován pomocí mikroelektronických a mechanických částí. Znamená to vývoj všech elektrických obvodů, desek tištěných spojů, obvodů pro unifikaci signálu nebo použití již hotových modulů. Model inteligentního domu by měl simulovat některé vybrané aplikace, které poskytují reálné inteligentní domy. Může tedy obsahovat tlačítko a LED diodu, které budou simulovat zvonek, webkameru, která by snímala prostor kolem vchodu do domu, motorky simulující dálkové otevírání brány a garážových vrat nebo stahování rolet, dále zabezpečení alarmem, který by mohl být realizován snímačem pohybu či světelnou závorou, nebo LED diody znázorňující jiné činnosti, které by se daly v domě automatizovat. Simulace by se mohla týkat také domovní elektrické sítě. Mohly by být dálkově vypínány jednotlivé zásuvky, rozsvěcovány světla popřípadě zatahovány rolety. Indikaci alarmu po otevření a zavření dveří je možné uskutečnit pomocí jazýčkového relé. Důležitá je také v každých inteligentních domech regulace teploty. Regulaci lze simulovat pomocí žárovky a teplotu v okolí by snímal teplotní snímač. V neposlední řadě by zde mohl být akumulátor, který by napájel dům v případě výpadku energie.

Popsané komponenty popřípadě řada dalších bude řízena prostřednictvím dostupného PLC a PIC. Oba řídicí systémy nad sebou budou mít nadřazený systém Control Web 6, kde bude vytvořena vizualizace jednotlivých aplikací. Pomocí této aplikace bude možno model také dálkově ovládat. Dále budou definovány alarmy v případě, že budou narušeny například dveře, pokud nebude nikdo doma.

Jako řídicí systém bude využíváno PLC řady SIMATIC S7-300. CPU má označení 314C-2DP. Je k dispozici 24 digitálních vstupů, 16 digitálních výstupů, 5 analogových vstupů a 2 analogové výstupy.

Další parametry:

- Profibus DP
- MPI rozhraní
- 1 vstup pro PT100
- rychlé čítače (60 kHz)
- Pracovní paměť 192kB

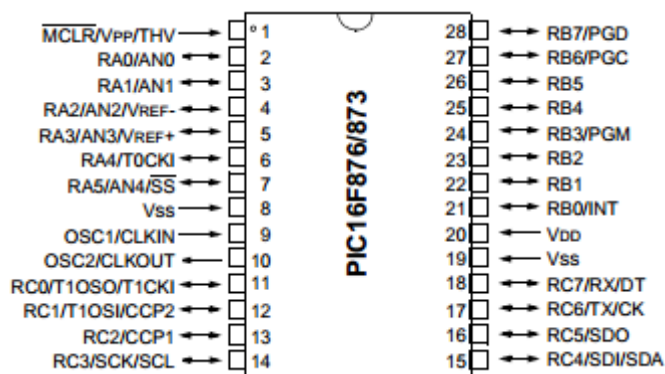


Obrázek 11 - Simatic S7 314C-2DP

Druhým použitým řídicím systémem je jednočipový mikroprocesor firmy Microchip typu PIC16F873A. Tento jednočipový mikroprocesor má 28 pinů, z nichž lze nastavit 22 jako vstupně/výstupní piny, které využívají vstupně/výstupní brány A, B, C. Tento mikroprocesor využívá sadu 35 instrukcí.

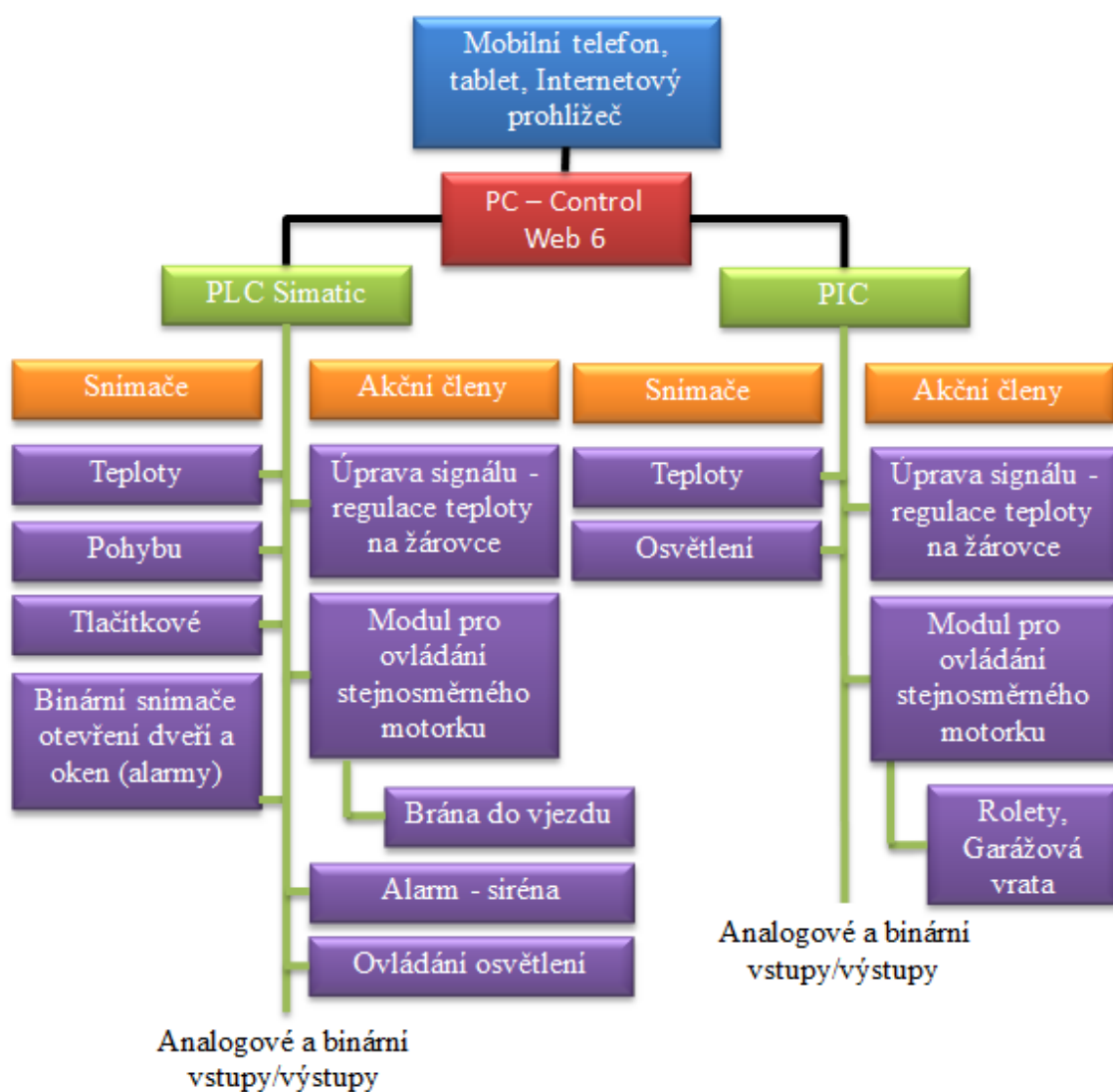
Tabulka 2 - Parametry PIC16F873A

Parametry	
Taktovací frekvence:	až 20 MHz
Velikost paměti programu Flash (14 bitových slov):	4kB
Velikost paměti:	192B
Velikost paměti EEPROM:	128B
Počet dostupných přerušení:	13
Časovače:	3
Moduly PWM:	2
Sériová komunikace:	USART, I2C, SPI
Paralelní komunikace:	neobsahuje
10-bit AD převodník:	5 vstupních kanálů
Komparátory:	2
Rozsah teplot:	-40...120 °C
Rozsah napětí:	2...5,5 V
Počet pinů:	28



Obrázek 12 - PIC16F873A s popisem pinů, viz[MICROCHIP TECHNOLOGY INC. 2012]

Nyní bude proveden návrh modelu inteligentního domu s využitím výše popsanych řídicích systémů a komponent.

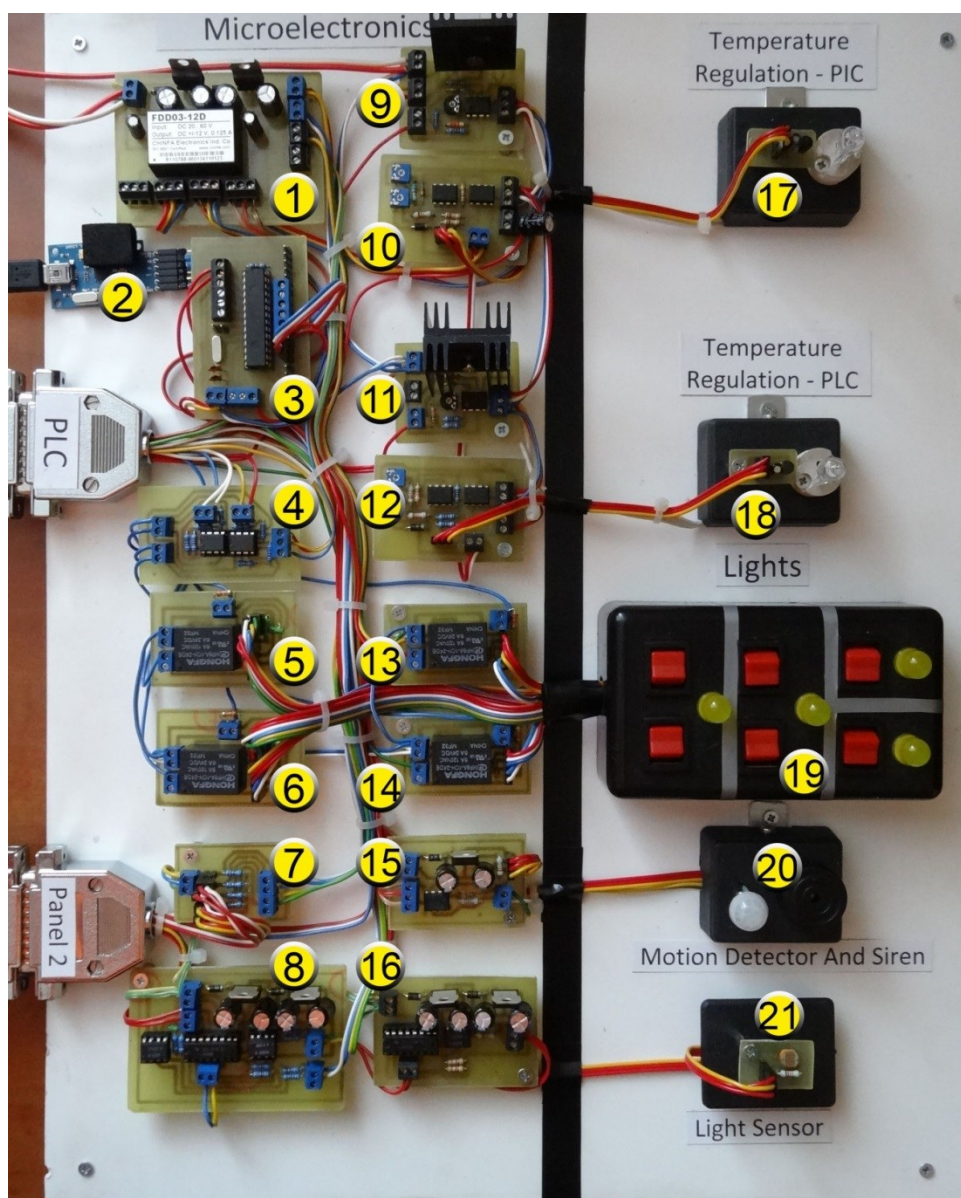


Obrázek 13 - Návrh modelu inteligentního domu

7 Realizace dílčích částí modelu

V následující části budou popsány dílčí části inteligentního domu, které byly na základě předcházejícího návrhu realizovány. Jedná se o návrh a zhotovení desek plošných spojů, které budou simulovat funkci regulace teploty, ovládání stejnosměrného motorku, který se dá využít pro simulaci otevírání brány nebo zatahování rolet, ovládání osvětlení a zabezpečovací systém. Tištěné spoje jsou navrženy v programu Eagle.

Všechny plošné spoje a části, realizující simulaci funkcí inteligentního domu, jsou umístěny na prvním panelu. Na druhém panelu je jen tištěný spoj pro úpravu signálu a napájení světelných závor, což bude popsáno později.



Obrázek 14 – Panel s vytvořenými mikroelektronickými obvody

Popis jednotlivých prvků na prvním panelu podle obrázku nahoře:

1. Napájecí modul
2. USB Serial Light Adapter
3. Modul s jednočipovým mikroprocesorem PIC16F873A
4. Modul pro zesílení signálu sloužícího pro indikaci světla
5. Modul simulující schodišťový vypínač – 2 tlačítka a relé
6. Modul simulující schodišťový vypínač – 2 tlačítka a relé
7. Napájení a indikace magnetických senzorů.
8. Modul pro ovládání 2 stejnosměrných motorků
9. Modul upravující napětí z PIC pro ovládání žárovky
10. Modul pro úpravu analogového signálu ze snímače teploty pro PIC
11. Modul upravující napětí z PLC pro ovládání žárovky
12. Modul pro úpravu analogového signálu ze snímače teploty pro PLC
13. Modul pro ovládání osvětlení – 1 tlačítko a relé
14. Modul pro ovládání osvětlení – 1 tlačítko a relé
15. Modul pro napájení a úpravu signálu z pohybového čidla PIR
16. Modul pro ovládání stejnosměrného motorku
17. Snímač teploty a žárovka simulující dvoupolohovou regulaci teploty
18. Snímač teploty a žárovka simulující PID regulaci teploty
19. LED diody a tlačítka simulující domovní osvětlení
20. Snímač pohybu PIR a sirénka
21. Snímač osvětlení

Dále jsou na panelu konektory pro napájení 24V a 12V, konektor na propojení panelu s PLC a konektor na propojení tohoto panelu s druhým. Nyní bude na schématu popsána logická struktura a vzájemná kooperace jednotlivých modulů. Bude zde uvedeno napájení, vstupní signály a výstupní signály, nebo který řídicí systém se k jakému modulu vztahuje.



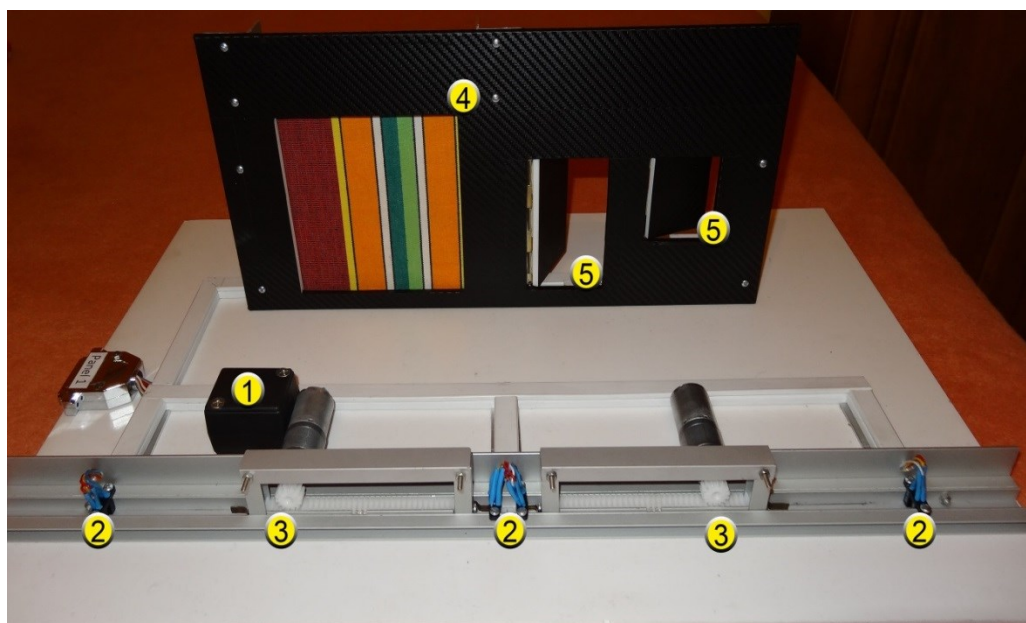
Obrázek 15 - Funkcionalita a kooperace jednotlivých modulů

Druhý panel, jak už bylo zmíněno, obsahuje tištěný spoj pro napájení a indikaci světelných závor. Tyto světelné závory slouží jako koncové snímače polohy brány, která je ovládána dvěma stejnosměrnými motorky. Brána se skládá ze dvou křídel. Konstruktivně byla brána sestavena pomocí ozubeného kola a ozubeného hřebene, který jezdí v kolejnici

profilu U. Dále je na druhém panelu další stejnosměrný motorek simulující stahování rolet nebo otevírání garážové brány a také dva magnetické senzory, indikující otevření dveří a okna. Na tomto panelu nejsou žádné tlačítka, a tudíž lze motorky ovládat pouze z aplikace vytvořené v Control Webu 6. Aplikace bude popsána v dalších kapitolách.

Popis jednotlivých prvků na druhém panelu podle obrázku dole:

1. Modul pro napájení a indikaci světelných závor
2. Světelné závory
3. Stejnosměrné motorky posunující křídly brány
4. Stejnosměrný motorek otevírající garážové vrata nebo stahující rolety
5. Magnetické senzory indikující otevření dveří a okna

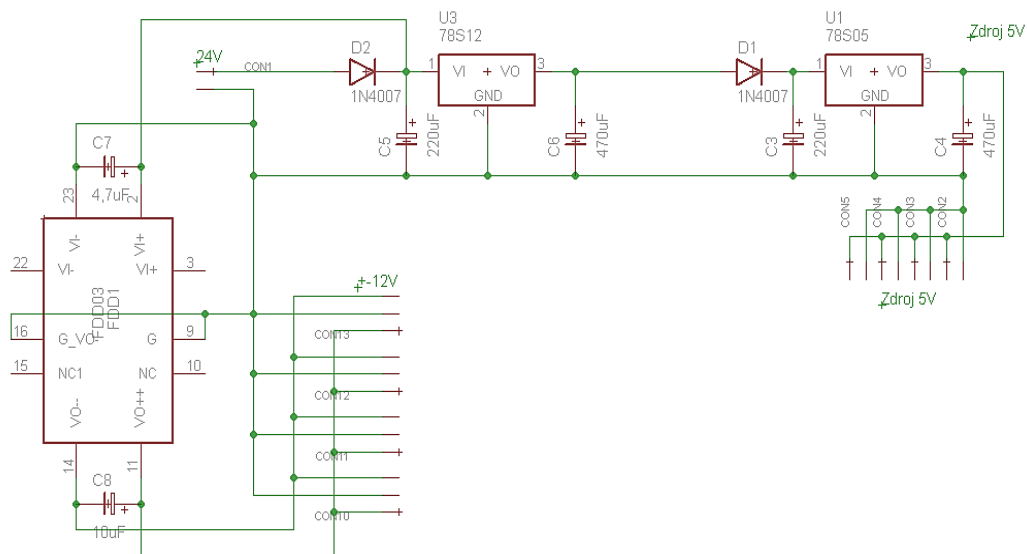


Obrázek 16 - Druhý panel simulující otevírání brány, rolety a indikaci otevření dveří a okna

Následující podkapitoly budou popisovat jednotlivé tištěné spoje, které byly zhotoveny.

7.1 Napájecí modul

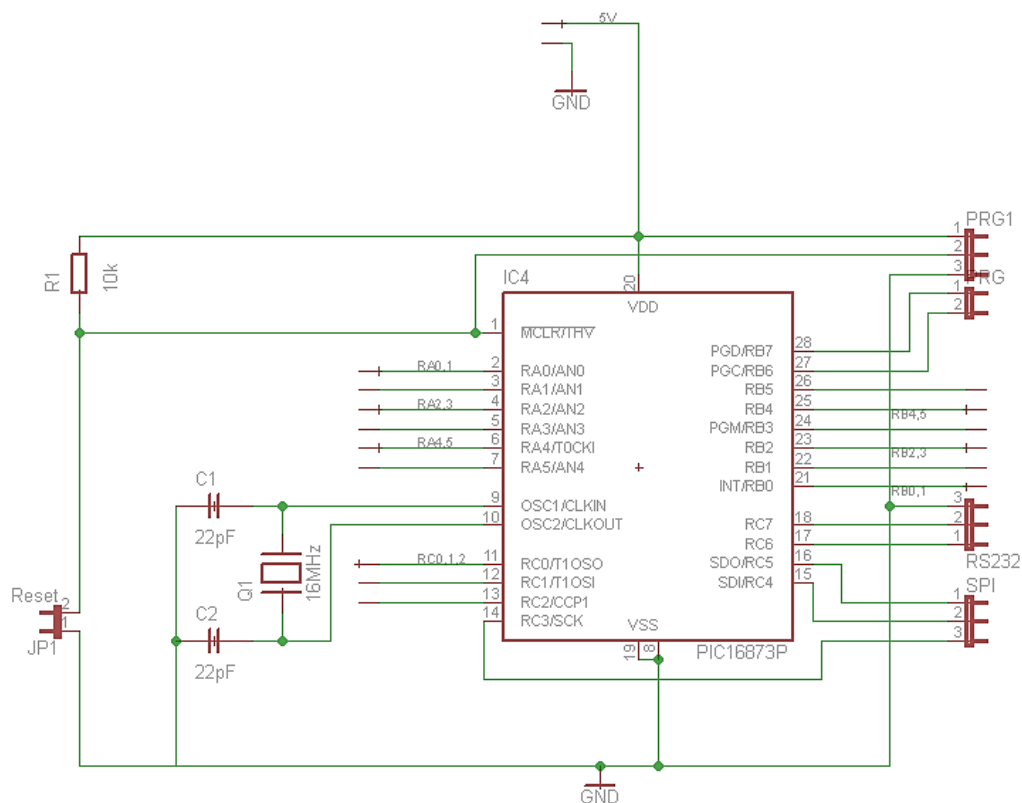
Prvním navrženým tištěným spojem byl napájecí modul. Na vstup je přivedeno napětí 24V z napájecího síťového adaptéru. Výstupní proud adaptéru je 2,7A a výstupní výkon 65W. Pomocí dvou stabilizátorů bylo stabilizováno napětí na 12V a 5V. Dále je na desce obsažen DC/DC měnič, který transformuje 24V na $\pm 12V$, které slouží jako napájení operačních zesilovačů na dalších tištěných spojích. Výstupní napětí 5V je pak distribuováno ostatním modulům.



Obrázek 17 - Schéma zapojení napájecího modulu

7.2 Rozhraní jednočipového mikroprocesoru

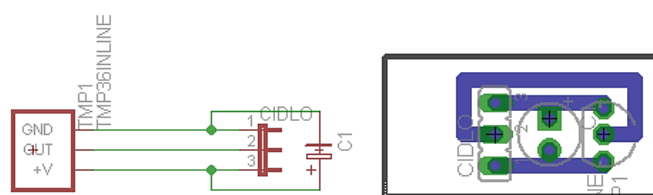
Tištěný spoj byl navržen tak, aby bylo možno využívat všechny piny a je tedy univerzální. Zapojení bylo navrženo podle manuálu. Byl zde použit 16MHz krystal a dva keramické kondenzátory.



Obrázek 18 - Zapojení PIC16F873A

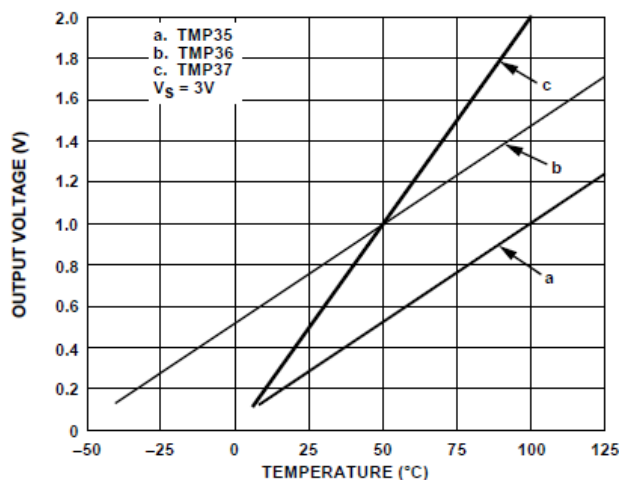
7.3 Snímač teploty TMP36GT9Z

Tištěný spoj pro tento snímač obsahuje pouze 3 součástky. Bylo zvoleno vytvořit jej zvlášť proto, že jej pak lze umístit kamkoliv dále od ostatních větších modulů. Tento snímač umožňuje měřit teplotu v rozsahu od -40°C do 125°C . Kondenzátor je umístěn na desce se snímačem proto, že odstraňuje vliv šumu a měl by být umístěn co nejblíže k snímači. Při teplotě 25°C je na výstupu 750mV . Při změně o $\pm 1^{\circ}\text{C}$ se napětí změní o $\pm 10\text{mV}$. Tento tištěný spoj je vyroben dvakrát, jelikož je použit u snímání a regulace teploty u PIC i PLC.



Obrázek 19 - Schéma zapojení a tištěný spoj teplotního snímače TMP36GT9Z

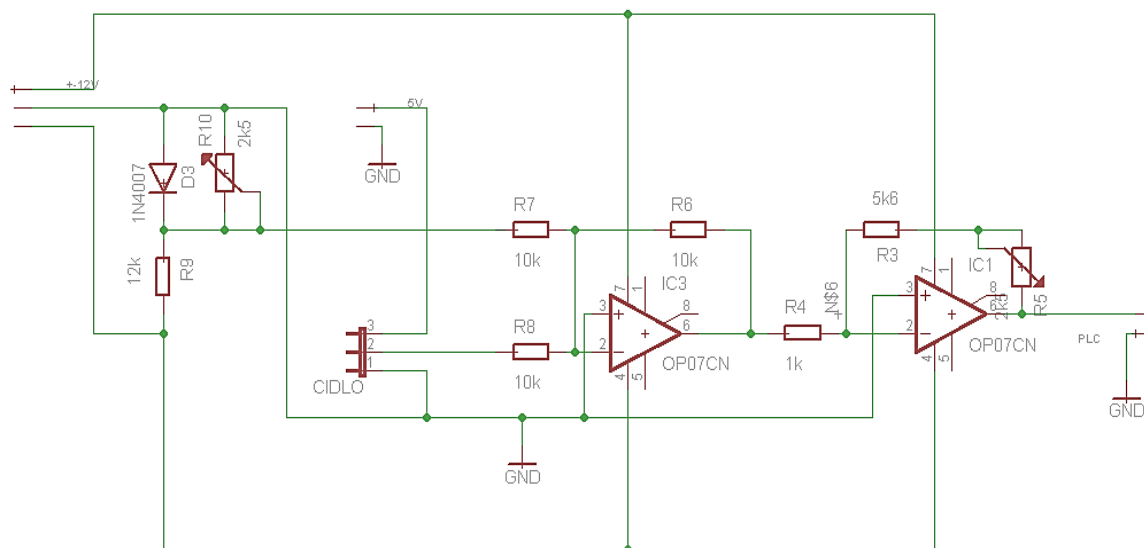
Výstupní napětí v závislosti na teplotě je lineární, což dokládá následující průběh.



Obrázek 20 - Závislost výstupního napětí na teplotě – průběh b, viz[ANALOG DEVICES 2005]

7.4 Modul pro úpravu signálu ze snímače

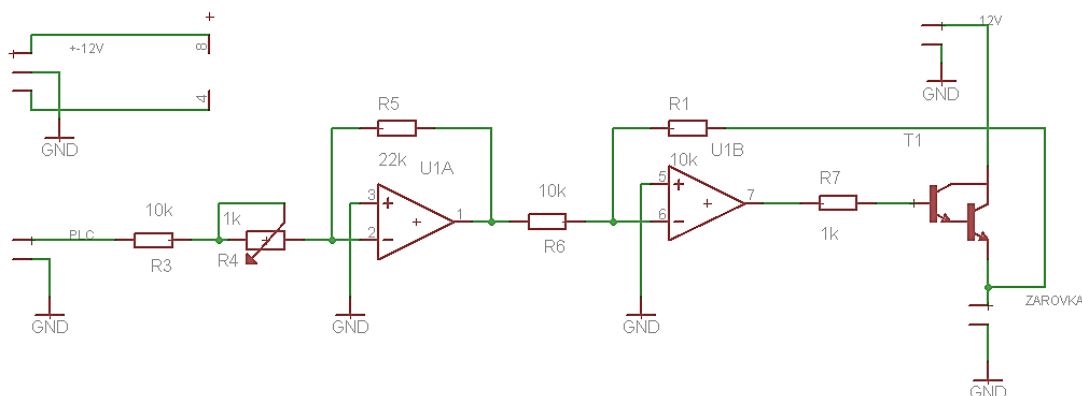
Jelikož lze na analogový vstup mikroprocesoru přivést napětí 0-5V, bylo nutné upravit signál ze snímače. Teplotní rozsah byl stanoven na -5°C až 80°C , čemuž odpovídá rozsah napětí 0,45V až 1,3V. Na vstupu bude tedy sumátor, kde se od výstupního napětí ze snímače, jehož minimum bylo stanoven na 0,45V, odečte napětí 0,45V. Po odečtení bude rozsah napětí od 0V do 0,85V. Aby bylo přibližně dosaženo napětí 5V v případě PIC a 10V v případě PLC, bude pomocí druhého operačního zesilovače OP07 napětí zesíleno.



Obrázek 21 - Schéma zapojení obvodu pro úpravu signálu ze snímače pro PIC

7.5 Modul pro ovládání žárovky

Následující modul bude realizovat ovládání žárovky. Použitá žárovka je na 12V, které se odebírají z napájecího adaptéru.

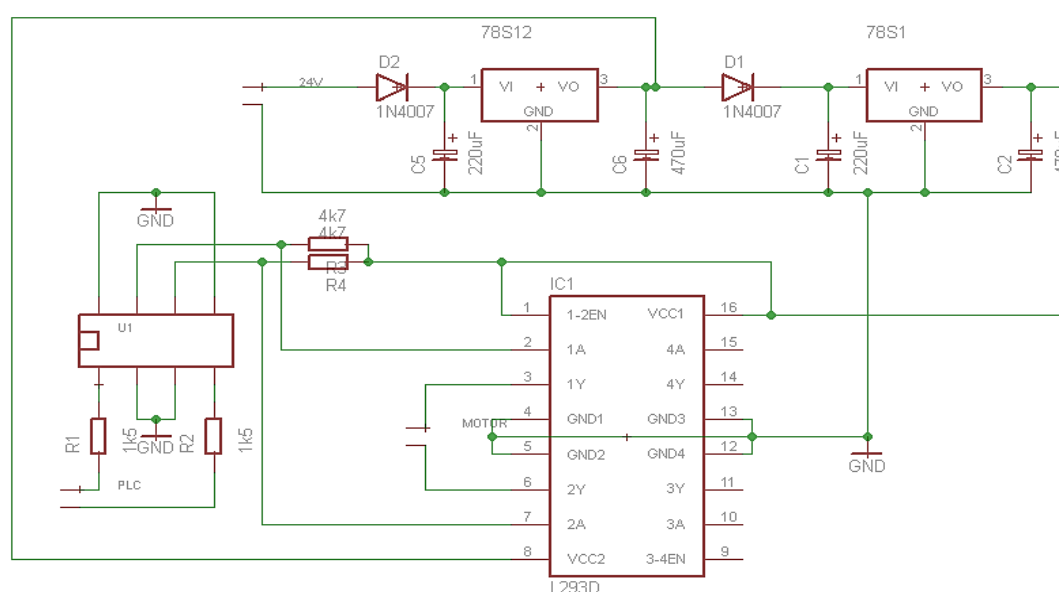


Obrázek 22 – Schéma zapojení pro ovládání žárovky z PIC

Řízení je realizováno pomocí tranzistoru TIP121. Je využito dvou operačních zesilovačů. První z nich zesiluje výstupní napětí z PIC a druhý jej invertuje a jako zpětná vazba slouží napájecí napětí na žárovce. Je tedy možno ovládat osvětlení žárovky spojitě. Jelikož je na tranzistoru velký úbytek napětí, nepodařilo se na výstup nastavit přesně 12V, ale maximálně 10V. Funkce regulace teploty tím ale není nijak ovlivněna, proto se tento úbytek může zanedbat. Tato deska je taktéž použita dvakrát a to pro úpravu signálu z PIC nebo PLC. Liší se jen hodnotami rezistorů, které poměrem udávají, jak má být zesíleno napětí z řídicího systému.

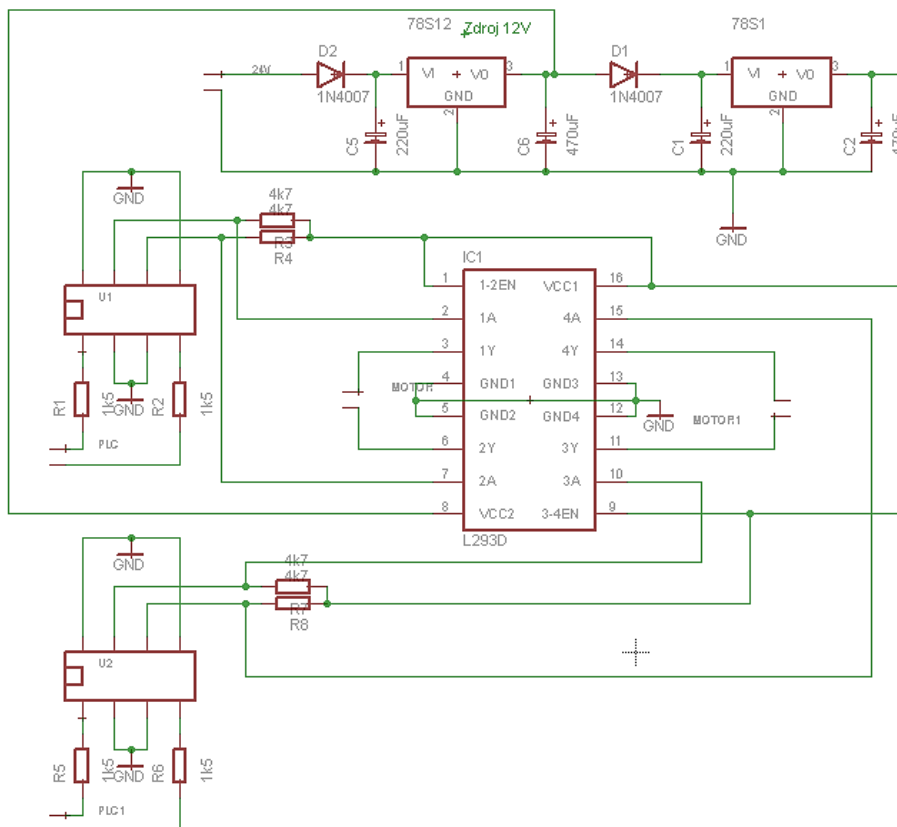
7.6 Moduly pro ovládání DC motorku

Tyto desky plošných spojů budou sloužit pro ovládání stejnosměrných motorků GM25-370CHV-130-R , napájené 12 Volty. První dva slouží pro simulaci otevírání brány a budou ovládány pomocí PLC. Třetí motorek bude ovládán pomocí jednočipového mikroprocesoru a je určen pro simulaci otevírání garážových vrat nebo stahování rolet. Pro řízení těchto motorků je využíváno součástky L293D, která obsahuje dva úplné H-můstky. Funkce H-můstku je taková, že dovede přepínat polaritu napájecího napětí na vstupu motorku a tím měnit směr otáčení.



Obrázek 23 - Schéma zapojení desky pro ovládání DC motorku – využití 1 H-můstku

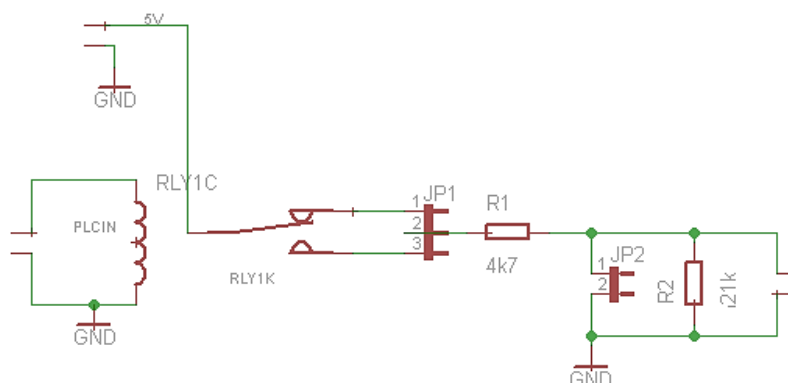
Jelikož bude tento modul ovládán z PLC, je nutno zajistit úpravu signálu z 24V na 5V. To je realizováno pomocí optočlenu s tranzistorem PC829.



Obrázek 24 - Schéma zapojení desky pro ovládání DC motorku – využity oba H-můstky

7.7 Moduly pro ovládání osvětlení

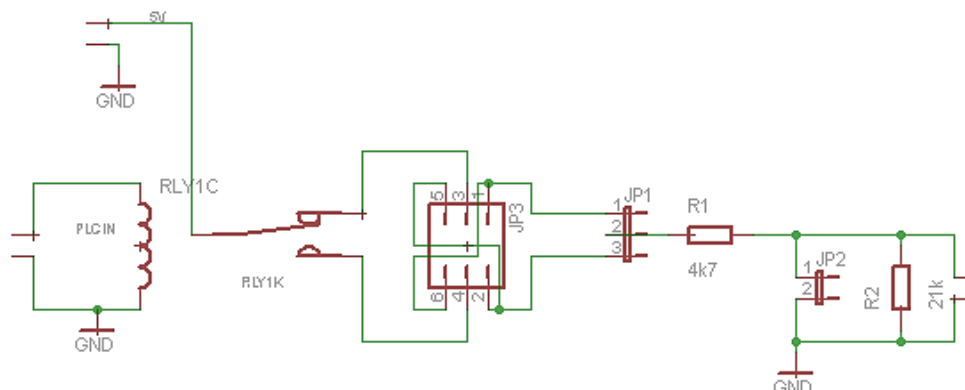
Tyto moduly jsou navrženy ve dvou verzích. U první verze je využíváno jednoho tlačítka a relé a u druhé verze dvou tlačítek a relé. Na vstupu těchto modulů je napětí 5V. Signál z PLC, který spíná relé, je 24V. LED diody, které simulují osvětlení, a tlačítka jsou připojeny na svorky.



Obrázek 25 - Schéma zapojení modulu pro ovládání osvětlení 1 tlačítkem

Zpětná vazba, zda je LED dioda rozsvícená či zhasnutá, je měřena na rezistoru, který je k LED diodě paralelně. Odpor 21kΩ je dost velký z toho důvodu, že napětí na diodě při

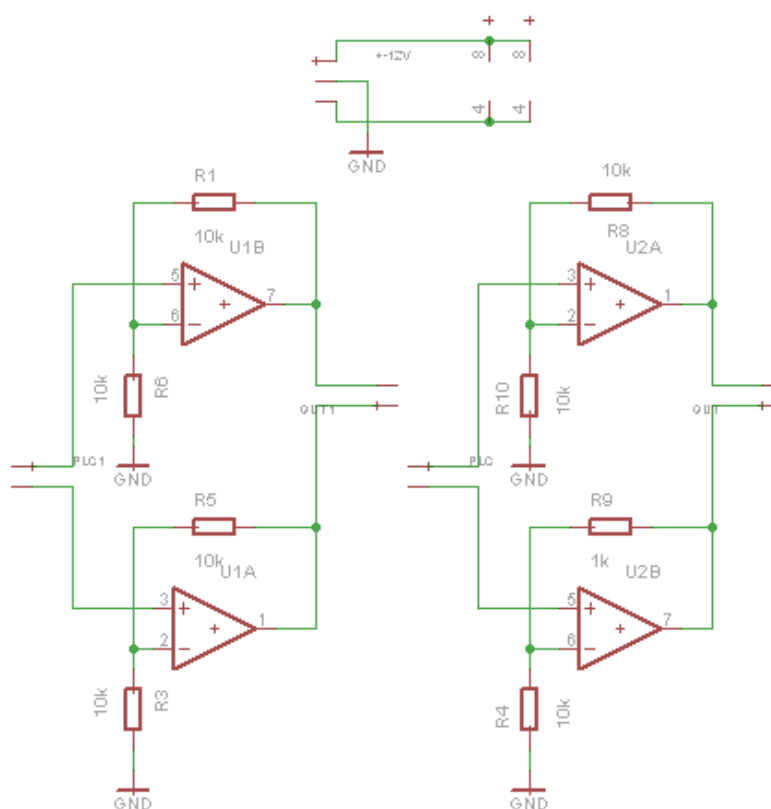
rozepnutém stavu bez tohoto odporu kolísalo kolem nulové hodnoty, což způsobilo při pozdějším zesílení tohoto signálu napětí dostatečně velké, aby jej PLC mohlo zaznamenat.



Obrázek 26 - Schéma zapojení modulu pro ovládání osvětlení 2 tlačítky

7.8 Modul pro unifikaci napětí z LED diod

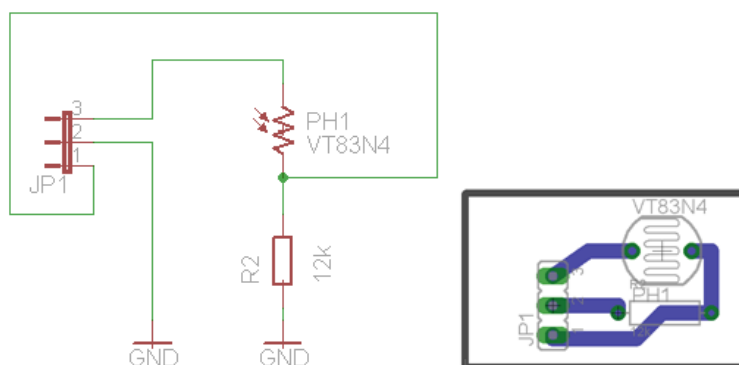
Vstupem do této desky plošných spojů jsou signály z LED diod z předchozích modulů pro ovládání osvětlení. Tyto signály jsou zde zesíleny pomocí operačních zesilovačů a poměru rezistorů na hodnotu, kterou dokáže PLC přečíst.



Obrázek 27 - Schéma zapojení modulu pro zesílení signálu z LED diod

7.9 Snímač osvětlení

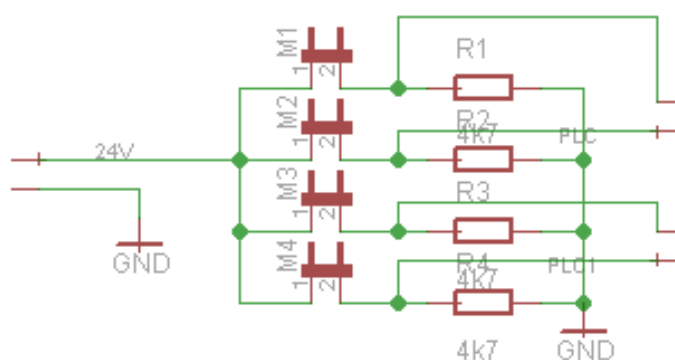
Jako snímač osvětlení byl využit fotorezistor VT83N4. Je využito jednoduchého zapojení děliče napětí, kdy se při změně osvětlení mění napětí na pevně daném odporu. Vstupní napětí je 5V. Výstupní napětí se tedy mění podle změny odporu fotorezistoru v rozsahu 0-5V, přičemž 0V je při úplné tmě.



Obrázek 28 - Schéma zapojení a deska plošných spojů snímače osvětlení

7.10 Napájení a indikace magnetických spínačů

Zapojení dveřních magnetických spínačů je velmi jednoduché. Na vstupu navržené desky tištěných spojů je napětí 24V a magnetické spínače jsou připojeny na svorky. Napětí, signalizující zda je spínač sepnutý, je přivedeno na svorky, které jsou propojeny s PLC.



Obrázek 29 - Návrh zapojení magnetických dveřních spínačů

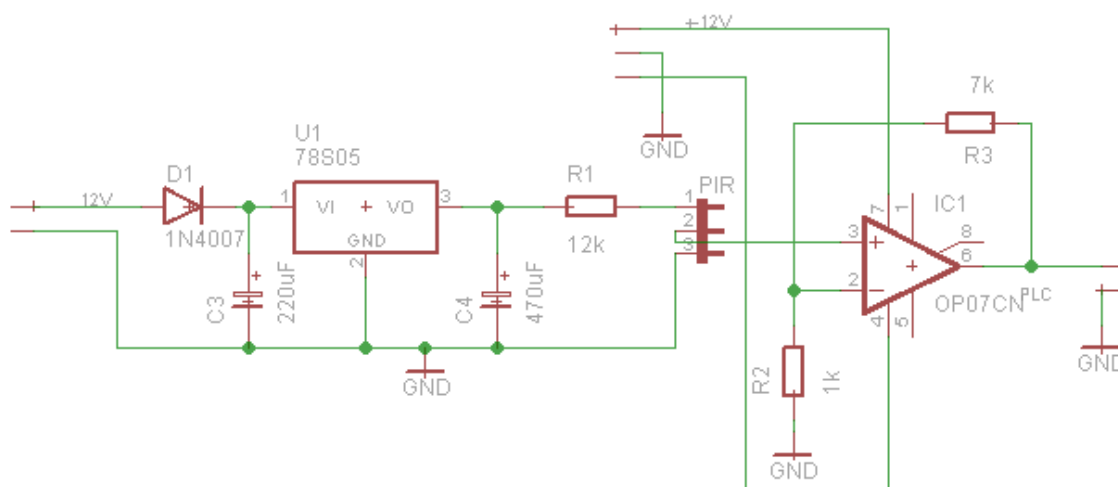
7.11 Napájení a indikace snímače pohybu

Modul slouží pro napájení snímače pohybu PIR SB00322A-1. Vstupní napětí je 12V, které je stabilizováno na 5V, což je napájecí napětí senzoru PIR. Vzdálenost detekce senzoru je 3 až 5 metrů a úhel detekce 100°.



Obrázek 30 - Čidlo pohybu PIR SB00322A-1

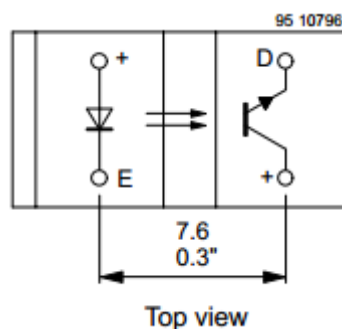
Jelikož je výstupní napětí čidla při logické jedničce 3V, je v zapojení také operační zesilovač OP07, který toto napětí zesílí tak, aby bylo možno jej zaznamenat z PLC.



Obrázek 31 - Schéma zapojení upravující signál z PIR senzoru

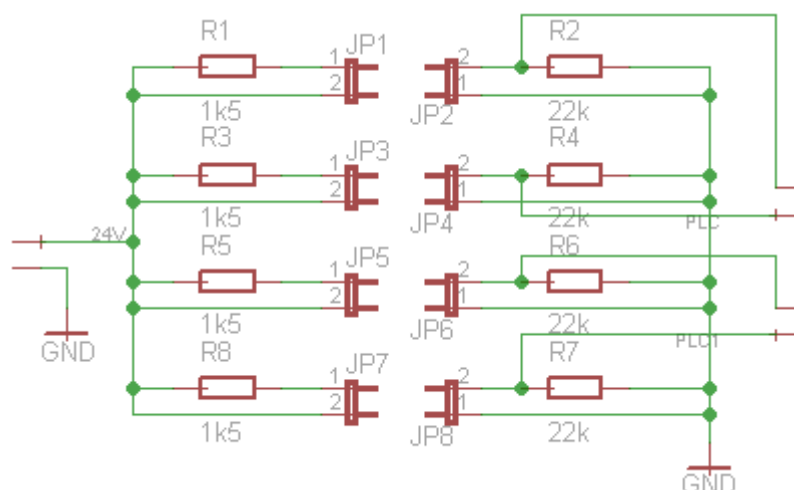
7.12 Napájení a indikace světelných závor

Světelné závory indikující pozici brány jsou typu TCST2103. Vnitřní uspořádání světelné závory je na obrázku.



Obrázek 32 - Vnitřní uspořádání světelné závory, viz [VISHAY 1999]

Na schématu lze vidět dvě řady svorek. Na levé straně je napájecí napětí 24V, přičemž napájení diody je připojeno přes rezistor 1k5. Na pravé straně je na svorkách nahoře kontakt D a dole E. Výstupní napětí je pak vyvedeno do PLC.



Obrázek 33 - Schéma zapojení světelných závor

8 Programování PIC

Zdrojový kód pro PIC16F873A byl napsán ve vývojovém programu mikroC. Nyní budou popsány jednotlivé části programu realizujícího čtení analogových veličin teploty a osvětlení, dvupolohovou regulaci teploty, ovládání stejnosměrného motorku a zprostředkovávajícího sériovou komunikaci s prostředím Control Web 6.



Obrázek 34 - Základní struktura programu v PIC

Cyklus běžící v nekonečné smyčce je prováděn každých 100ms. Tento skenovací cyklus je nastaven stejně tak i v PLC.

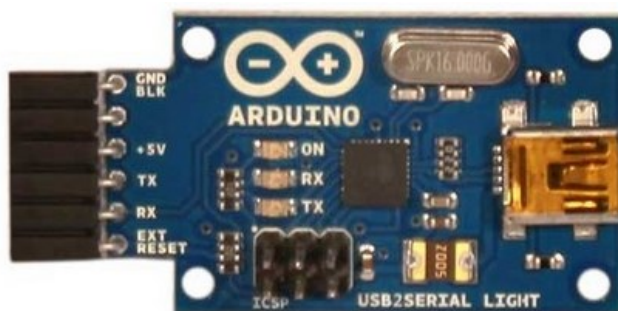
8.1 Čtení analogové hodnoty

Pro čtení analogové hodnoty bylo potřeba dvou funkcí, a to pro inicializaci analogového vstupu a pro přečtení analogové hodnoty. PIC16F873A využívá pro převod analogové hodnoty 10-bitový převodník, což znamená, že pro rozsah 0V – 5V lze mít rozlišitelnost $5/1023 \approx 0,005V$. Popis nastavení registrů pro čtení analogové hodnoty je v příloze.

Do registrů ADRESH a ADRESL se ukládá 10-bitový výsledek A/D převodu. V programu je využito zarovnání vpravo, což znamená, že v registru ADRESL bude uloženo 8 bitů a v ADRESH zbylé 2 bity. Ve funkci ADC_Read, která čte analogovou hodnotu, je realizován převod, který vrací jako výstup z funkce přímo hodnotu typu „unsigned int“. Tato funkce je obsažena přímo v knihovně ADC_Library.

8.2 Sériová komunikace se systémem Control Web 6

Program v jednočipovém mikroprocesoru komunikuje s aplikací v Control Web 6 pomocí sériové komunikace. Je využito převodníku *Arduino Serial Light Adapter*, který převádí kanály Rx a Tx na USB rozhraní.



Obrázek 35 - Arduino Serial Light Adapter

V každém cyklu programu se do aplikace v Control Web 6 posílají aktuální hodnoty z A/D převodu. Ty jsou uloženy v jednom řetězci, který se v Control Web 6 opět rozdělí a přiřadí vlastním proměnným. Dále se program ptá, zda nepřišel nějaký znak. Pokud ano, kontroluje se jaký. Je zde možno přechíst 9 znaků, na jejichž základě se vykoná nějaká akce:

Tabulka 3 - Akce následující po přijetí znaku

Přijatý znak	Následující akce
*	<pre> if (uart_rd == '*'){ UART1_Read_Text(Input,"###",255); Hyst[0]=Input[4]; Hyst[1]=Input[5]; SP[0]=Input[1]; SP[1]=Input[2]; SP[2]=Input[3]; SetPoint = atoi(SP); H = atoi(Hyst); uart_rd='\0'; } </pre>
1	<pre> if (uart_rd == '1'){ start=1; uart_rd='\0'; } </pre>
0	<pre> if (uart_rd == '0'){ start=0; uart_rd='\0'; } </pre>
S	<pre> if (uart_rd == 'S'){ start=0; SetPoint=310; H=14; uart_rd='\0'; if (poc == 51) poc=50; if (poc>0){ </pre>

Přijatý znak	Následující akce
U	<pre> Gate_down=1; }} if (uart_rd == 'U'){ Gate_down=0; Gate_up=1; if (poc == -1) poc=0; uart_rd='\0'; } </pre>
D	<pre> if (uart_rd == 'D'){ Gate_up=0; Gate_down=1; if (poc == 51) poc=50; uart_rd='\0'; } </pre>
I	<pre> if (uart_rd == 'I'){ Gate_up=0; uart_rd='\0'; } </pre>
F	<pre> if (uart_rd == 'F'){ Gate_down=0; uart_rd='\0'; }} </pre>

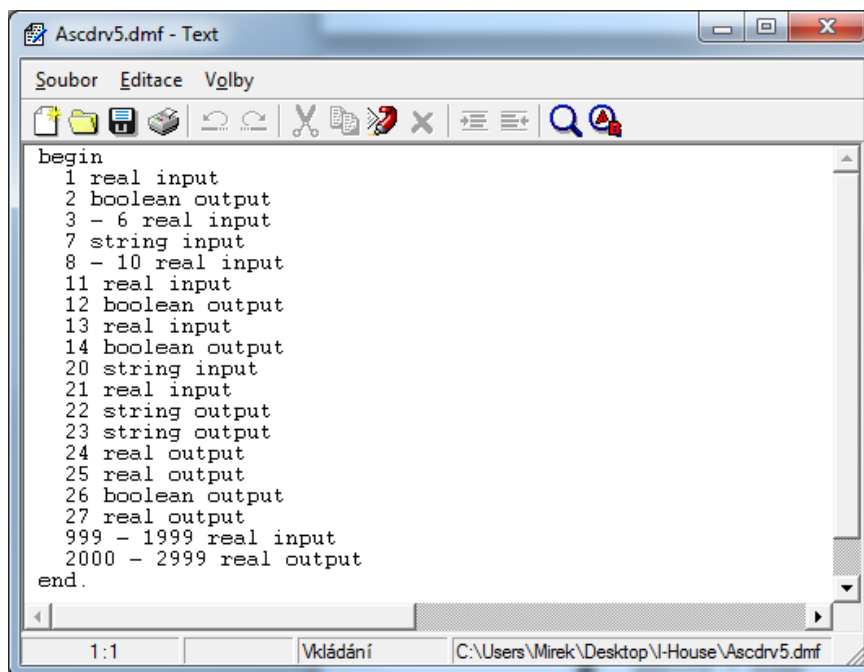
Pokud je tedy přijat znak „*“, čeká se, dokud se nepřijme celý řetězec, který je ukončen znaky „##“. Následně jsou znaky řetězců rozděleny do dalších dvou řetězců, které se převádí na čísla, která reprezentují žádanou veličinu a hysterezi. Pokud je přijatý znak „1“ nebo „0“, může být samotná dvupolohová regulace spuštěna nebo zastavena. Při spuštění aplikace v Control Webu se odešle znak „S“, který inicializuje všechny proměnné. Dalšími znaky jsou „U, D, I a F“, které mají za úkol spínat stejnosměrné motorky, simulující dvoukřídlovou bránu tak, aby se otáčely jedním nebo druhým směrem.

Na straně Control Webu je tedy využíváno ovladače zajišťujícího sériovou komunikaci jménem ASCDRV5. Pomocí ovladače ASCDRV5 je možno realizovat komunikaci se zařízeními pomocí sériového rozhraní RS-232, RS-485 nebo RS-422. Data se přenášejí ve formě textových řetězců – posloupnosti ASCII znaků ukončených jedním nebo několika znaky (terminátory). Pokud jsou data přijímána, ukládají se nejprve do příchozího zásobníku (bufferu). Po přijetí ukončovacího znaku se data z bufferu uloží do fronty. V tomto okamžiku se generuje výjimka od ovladače. Po zpracování výjimky se musí povolit nová zápisem na výstupní kanál. Data lze tedy číst jen po přijetí výjimky od ovladače, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010B]..

Pro odesílání dat existují dva režimy – synchronní a asynchronní. Ten musí být nejdříve nakonfigurován. V asynchronním režimu se čeká na odpověď od protějšího zařízení. Pokud odpověď nepřijde, je vygenerována chyba. V synchronním režimu lze

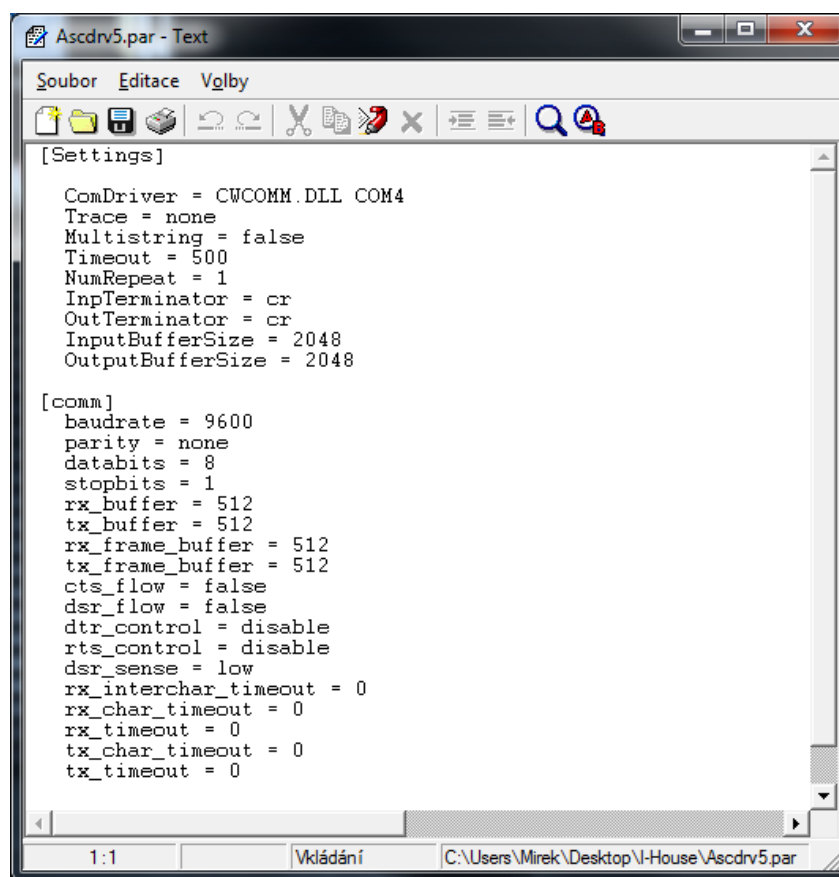
určit, jestli má být další řetězec odeslán ihned, nebo až po povolení výjimky od ovladače, viz [MORAVSKÉ PŘÍSTROJE a.s. 2010B].

Po vytvoření instance ovladače v datových inspektorech je nutné nakonfigurovat dva soubory – parametrický a mapovací.



Obrázek 36 - Mapovací soubor ASCII ovladače

V mapovacím souboru se definují typy kanálů ovladače. Ve vytvořené aplikaci je využíváno kanálu *CH_vyjimka*, který má index 2 a slouží pro nastavování výjimky od ovladače. Pro příjem dat je využíváno kanálu 20 a pro odesílání kanálu 22.



Obrázek 37 - Parametrický soubor ASCII ovladače

Parametrický soubor je důležitý pro nastavení komunikace. V sekci *Settings* lze nastavit například jméno komunikačního portu COMx nebo terminátory, kterými jsou ukončeny vstupní a výstupní řetězce. V sekci *Comm* lze nastavit rychlost komunikace, počet bitů, jestli je využíváno paritního bitu nebo počet start nebo stop bitů.

8.3 Algoritmy ovládání motorku a regulace teploty

Když už je funkční komunikace mezi SCADA/HMI systémem a jednočipovým mikroprocesorem, tak bylo nutno naprogramovat samotné algoritmy dvoupolohové regulace a ovládání stejnosměrného motorku, simulujícího otevírání a zavírání garážových vrat nebo zatahování rolet.

Jelikož u motorku nejsou využívány žádné snímače polohy, je tento motorek ovládán na základě času. Motorek navine roletu od spodu až nahoru za 5 sekund. Proto byla vytvořena proměnná, která se každý skenovací cyklus zvýší o 1 v případě, že motorek navíjí roletu nahoru, a sníží o 1 v případě, že se roleta odvíjí dolů. Tímto je zajištěno to, že na základě hodnoty proměnné víme, kolik skenovacích cyklů zbývá do koncové polohy.

Aby bylo zajištěno bezchybné pozice i v případě vypnutí aplikace v CW6, tak při startu nebo ukončení aplikace je motorek vrácen do inicializační polohy.

Co se týče algoritmu dvoupolohové regulace, tak bylo využíváno jednoduchých podmínek. Při překročení horní meze, v případě že je žárovka sepnutá, se akční člen (žárovka) vypne a při překročení spodní meze, v případě že je žárovka vypnutá, se sepne.

9 Konfigurace a programování PLC

V minulé kapitole byl popsán program vytvořený pro jednočipový mikroprocesor. Nyní tedy bude podrobněji popsán také program vytvořený v PLC Simatic 314C-2DP. V první části bylo nutno PLC připojit k PC a správně nakonfigurovat. To obnášelo správně nastavit hardwarovou konfiguraci, analogové a digitální vstupy a výstupy a jejich adresy, dobu skenovacího cyklu apod. Pro připojení PLC pomocí programátoru se tedy musela nejprve nastavit adresa ovladače „PCAdapter MPI“ v nastavení Simatic Manageru „Set PG/PC Interface“ a dále adresa v HW konfiguraci.

9.1 Program v PLC

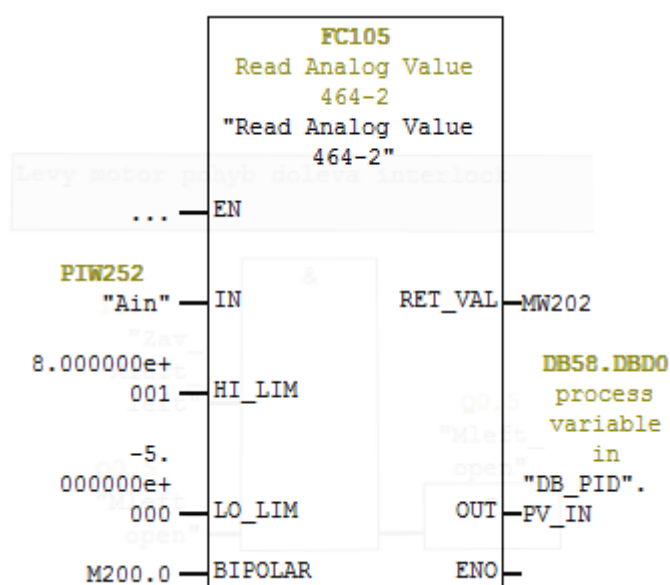
Druhým krokem při vytváření aplikace v PLC bylo nadefinování vstupních a výstupních proměnných a přiřazení jejich fyzických adres. To bylo provedeno v *Symbol Editoru*. Výpis všech proměnných byl sepsán v tabulce v Excelu, viz příloha.

Statu	Symbol	Address	Data type	Comment
9	MagSensor1	I 0.5	BOOL	Magnetic sensor 1 indication
1	MagSensor2	I 0.6	BOOL	Magnetic sensor 2 indication
1	Mleft_close	Q 0.6	BOOL	Left motor close command
1	Mleft_open	Q 0.5	BOOL	Left motor open command
1	Mright_close	Q 1.0	BOOL	Right motor close command
1	Mright_open	Q 0.7	BOOL	Right motor open command
1	CtrlLight1	Q 0.0	BOOL	Light 1 switch
1	CtrlLight2	Q 0.1	BOOL	Light 2 switch
1	CtrlLight3	Q 0.2	BOOL	Light 3 switch
1	CtrlLight4	Q 0.3	BOOL	Light 3 switch
1	PIR	I 0.4	BOOL	PIR sensor indication

Obrázek 38 - Některé definované proměnné v Symbol Editoru

Následovalo samotné programování. Pro ovládání stejnosměrných motorků byl vytvořen funkční blok FB1. Ten se stará o to, že při dosažení koncové polohy se motorky vypnou. Příkazy pro motorky jsou zadávány ze SCADA systému Control Web 6.

Dále je vytvořen a nakonfigurován funkční blok FB58 s názvem „TCONT_CP“, což je PID regulátor teploty. Na základě tohoto bloku byl vytvořen také datový blok DB58. Aby bylo možno tento blok využívat, bylo nutno zajistit cyklické volání tohoto regulátoru pomocí organizačního bloku OB35, který se stará o cyklické přerušení. Časová konstanta přerušení byla nastavena na 100ms. Dále byly nakonfigurovány analogové vstupy a výstupy na rozsah 0-10V. Do tohoto bloku byla vložena funkce SCALE zajišťující převedení výsledné hodnoty A/D převodníku na hodnotu zvoleného teplotního rozsahu, dále samotný regulátor a nakonec funkce UNSCALE, která slouží pro převedení hodnoty z regulátoru na analogový výstup.



Obrázek 39 - Příklad využití funkčního bloku SCALE

Při spuštění PLC je nutno regulátor inicializovat. Musí se přiřadit doba cyklu a restartovat se. K tomu je využit organizační blok OB100. Nastavování parametrů regulátoru a jeho výsledná funkcionality bude podrobněji popsána v popisu aplikace v Control Web 6. Ostatní proměnné jako jsou například indikace nebo ovládání světel slouží v PLC pouze jako rozhraní pro komunikaci mezi hardwarem a prostředím Control Web 6.

9.2 Komunikace mezi PLC a Control Web 6

Nyní bude popsáno, jakým způsobem je realizována komunikace mezi PLC Simatic a systémem Control Web 6. Byla zvolena komunikace pomocí protokolu OPC (OLE for Process Control) a to hlavně díky své jednoduchosti.

OLE for Process Control (OPC) představuje první úspěšnou iniciativu standardizující komunikační rozhraní mezi prvky průmyslové automatizace - průmyslovými automaty, čidly a akčními členy na jedné straně a řídicími či operátorskými počítači a průmyslovými informačními systémy na straně druhé. Ze standardu OPC profitují především uživatelé, kteří díky němu přestávají být vázáni na programové a technické vybavení podporující pouze vlastnické protokoly zavedených firem (zejména pokud tyto firmy brání své postavení právními překážkami zabraňujícími implementaci daného protokolu třetím stranám). Standard OPC spravuje nezisková organizace OPC Foundation a implementace tohoto standardu je dostupná všem bez jakýchkoliv licenčních poplatků, viz[MORAVSKÉ PŘÍSTROJE a.s. 2010B].

OPC standard definuje řadu způsobů komunikace, aby vyhověl různým požadavkům klientů:

- Synchronní komunikace vždy čekající na přenos dat z/do zařízení.
- Synchronní komunikace pracující s vyrovnávací pamětí serveru (cache).
- Asynchronní komunikace (vždy komunikuje se zařízením).
- Periodická komunikace serveru se zařízením a zpětné volání klienta při změně dat.

Druhý a čtvrtý způsob komunikace předpokládá, že server sám dokáže vyvolávat komunikaci s periferií a buď jen ukládat přečtená data do vyrovnávací paměti (odkud mohou být synchronně čtena) nebo předávat data klientovi zpětným voláním (jeden ze způsobů asynchronní komunikace), viz[MORAVSKÉ PŘÍSTROJE a.s. 2010B].



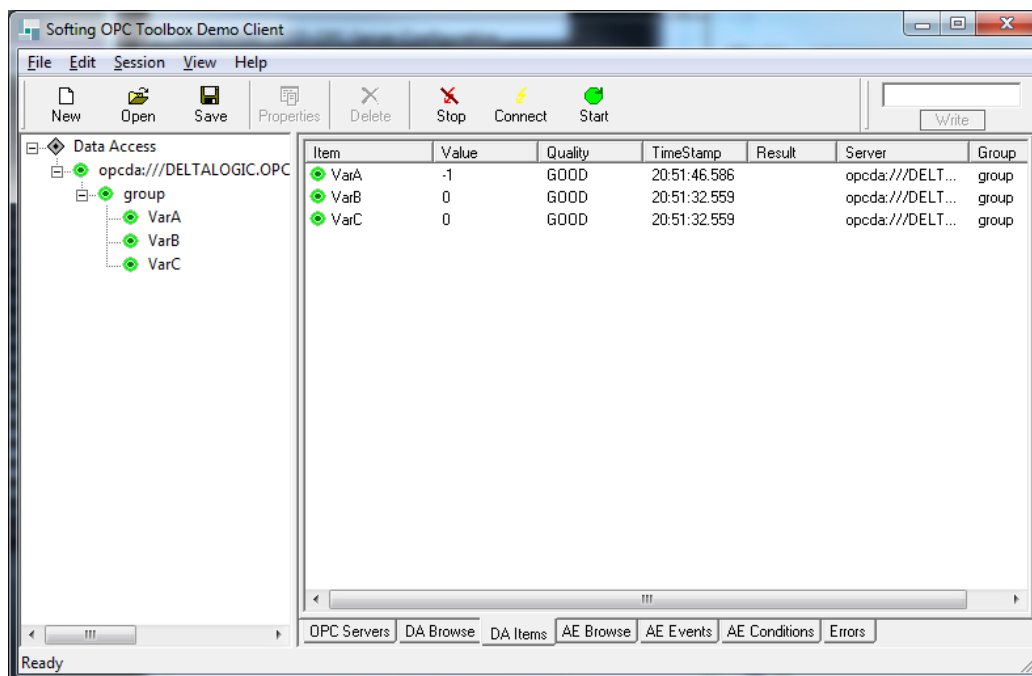
Obrázek 40 - Komunikace mezi PLC a Control Web 6

Komunikace mezi PLC a PC je realizována pomocí programátoru využívajícího MPI rozhraní. OPC komunikace tedy je nastavena stejně jako komunikace pro PLC v Simatic Manageru. Je využíváno softwaru s názvem Deltalogic S7/S5 OPC Server. Bylo nutno definovat proměnné pomocí textového souboru s přesně daným obsahem.

#	Alias Name	Syntax Name	Komentar
	VarA	i0.0	Pokus A - vstup
	VarB	i0.1	Pokus B - vstup
	VarC	q10.0	Pokus C - výstup

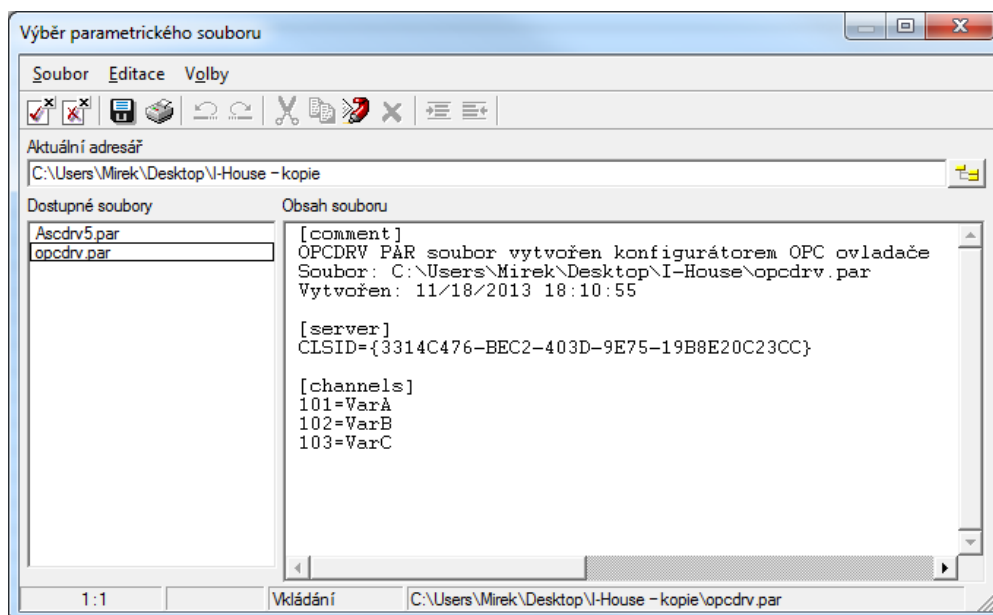
Obrázek 41 - Textový soubor definující proměnné pro OPC komunikaci

Sledovat stav proměnných pak lze přes Softing OPC Toolbox. Ten však běží v omezené verzi a po 90 minutách se musí restartovat.



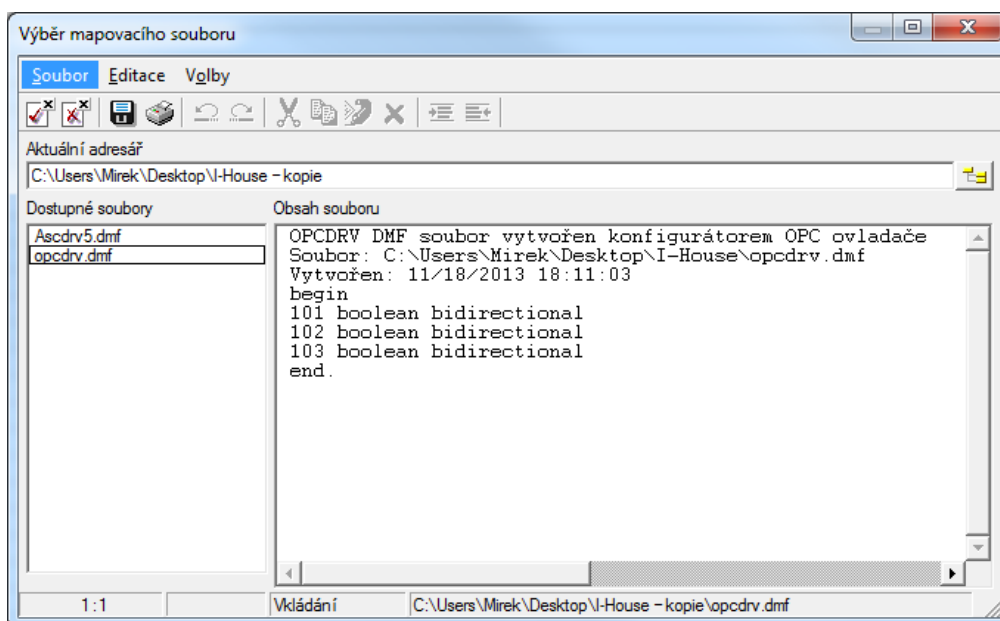
Obrázek 42 - Softing OPC Toolbox

Pro připojení Control Webu k OPC serveru se musí nejprve nastavit dva konfigurační soubory - parametrický a mapovací. Parametrický je rozdělen na jednotlivé sekce. Každá z nich začíná názvem v hranatých závorkách. V sekci *channels* jsou definovány kanály, použité pro komunikaci mezi PLC a Control Webem.



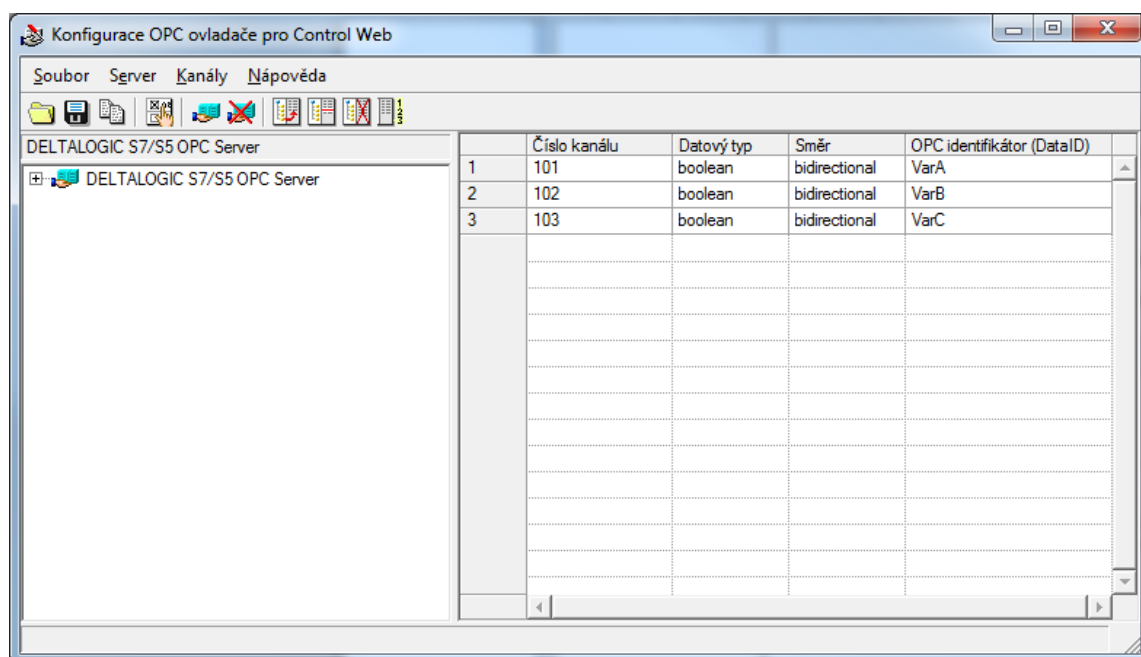
Obrázek 43 - Parametrický soubor OPC ovladače

Mapovací soubor určuje datový typ použitých kanálů a směr. Směr lze nastavit na input, output nebo bidirectional (vstup, výstup a vstup i výstup zároveň).



Obrázek 44 - Mapovací soubor OPC ovladače

Tyto soubory se vygenerují pomocí nástroje Control Webu. Zde se musí definovat pouze čísla kanálů.

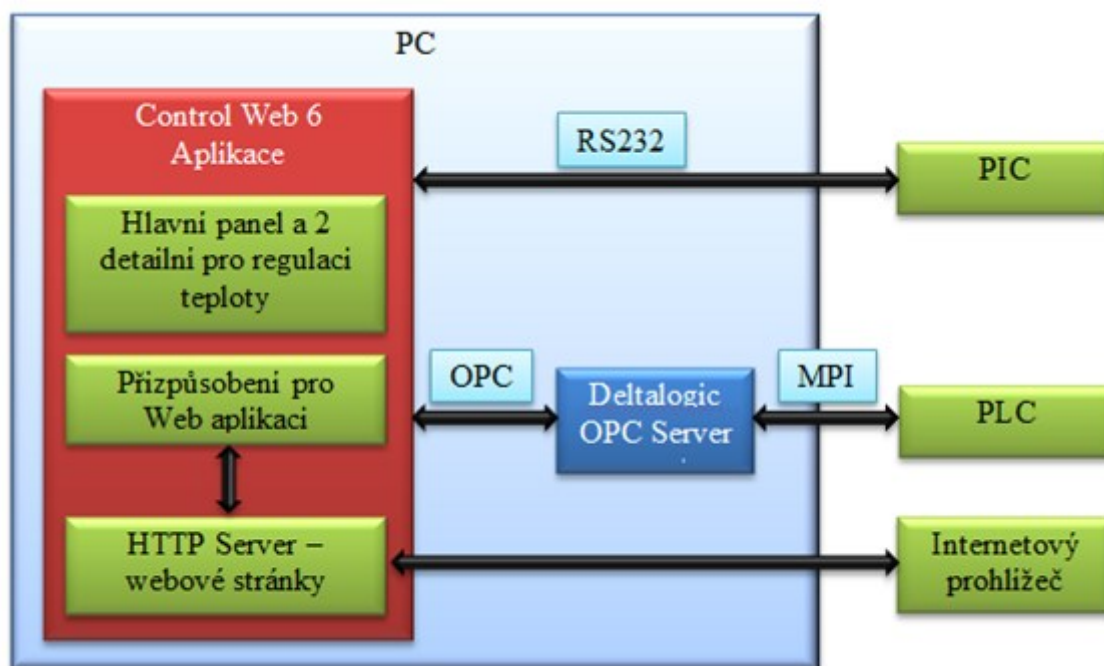


Obrázek 45 - Nástroj pro konfiguraci OPC serveru

Posledním krokem pro uvedení OPC komunikace do provozu bylo vytvoření nového ovladače v datových inspektorech. Zde se musel přiřadit již vygenerovaný mapovací a parametrický soubor. V Control Webu se pak mohou vytvářet proměnné typu „channel“, které slouží právě pro komunikaci.

10 Návrh aplikace v prostředí Control Web 6

Aplikace vytvořená v Control Webu 6 realizuje rozhraní mezi řídicím systémem a uživatelem. Jak již bylo popsáno dříve, Control Web 6 komunikuje s jednočipovým mikroprocesorem PIC pomocí sériové komunikace a s PLC pomocí OPC komunikace. Jelikož se jedná o aplikaci pro sledování a ovládání procesů modelu inteligentní domácnosti, bude také vytvořen interface, který se pomocí Control Webu a komponenty httpd vygeneruje do podoby webové stránky. Z té bude možno sledovat stav, v jakém se jednotlivé aplikace nachází, z prostředí Internetu nebo intranetu.



Obrázek 46 - Softwarová struktura aplikace v prostředí Control Web 6

Z hlediska bezpečnosti se musí zvážit, jakým způsobem lze vzdáleně komunikovat. V Control Webu lze nastavit uživatele a jejich hesla a pro připojení k webové aplikaci modelu inteligentního domu tyto identifikační údaje po uživateli požadovat, ale v reálném inteligentním domě by toto řešení určitě využití nemělo, protože tento způsob není příliš bezpečný. Proto je vhodné zvážit připojení například prostřednictvím VPN. Za takového stavu je komunikace šifrována a ověřována pomocí digitálních certifikátů.

10.1 Vzhled a funkce vytvořené aplikace

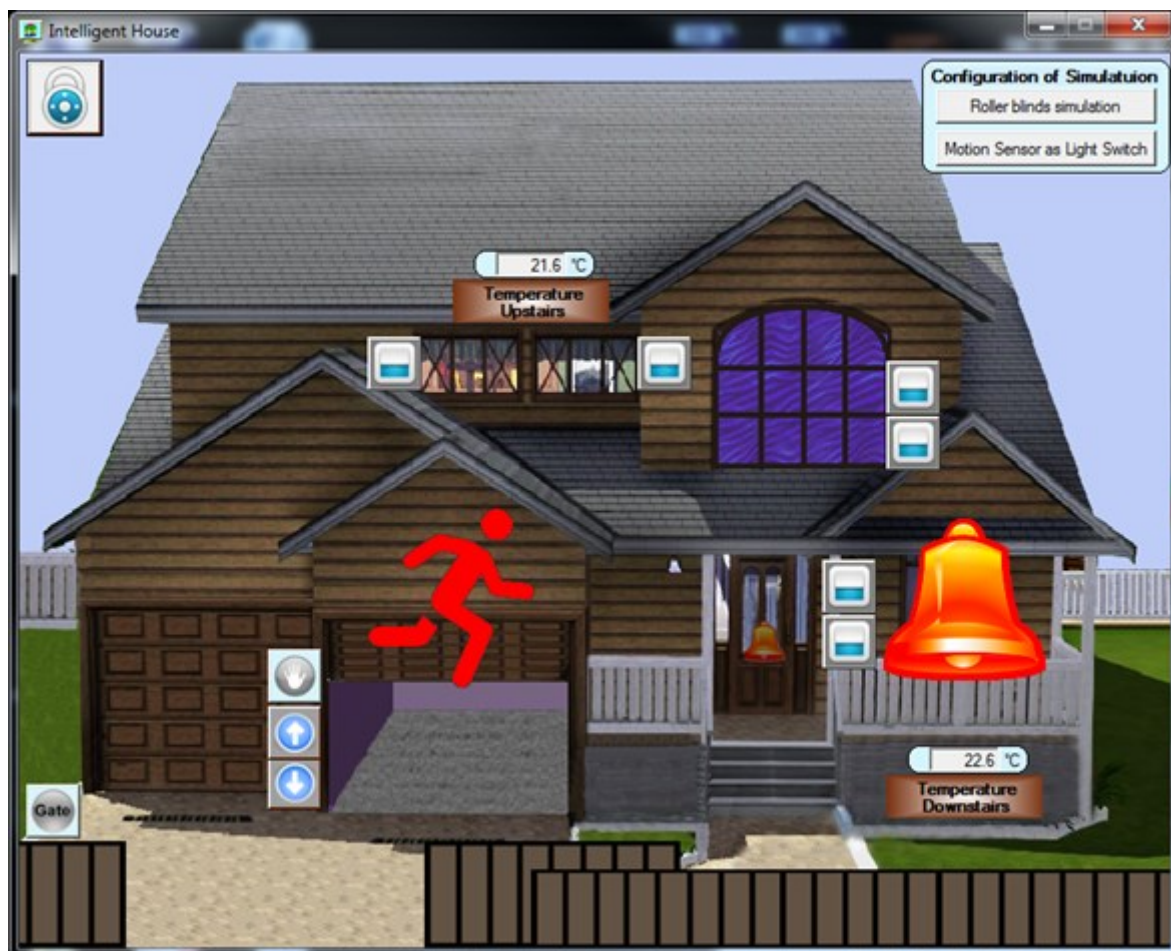
Vytvořená aplikace v Control Web 6 se skládá z hlavní obrazovky a dalších dvou panelů, které slouží pro konfiguraci parametrů regulace teploty. Na hlavní obrazovce je přehled všech důležitých částí, které se vztahují k jednotlivým částím modelu inteligentního domu.



Obrázek 47 - Vzhled aplikace vytvořené v Control Web 6

Jsou zde tlačítka, kterými se zapínají a vypínají světla. Pokud jsou světla zapnuta, barva v oknech se změní na žlutou. Indikují tedy jejich aktuální stav. Dále jsou zde zvony na spodním okně a dveřích. Pokud není aktivovaný zabezpečovací systém, který se aktivuje tlačítkem v levém horním rohu, a dveře nebo okno jsou otevřeny, mají zvony barvu zelenou. Pokud je vše zavřeno, barva této indikace je stejná jako na obrázku. V případě zabezpečeného domu a narušení se spustí alarm. Začne houkat fyzická sirénka a také se spustí alarm v aplikaci. Taktéž ikona zvonu symbolizující narušení se mnohokrát zvětší a začne blikat, aby bylo patrné, kde k narušení došlo. Stejnou funkci plní ikona pohybující se postavy nad garážovými vraty. Pokud se v garáži nic nepohybuje, má postava černou barvu. V případě pohybu při neaktivovaném zabezpečovacím systému je zelená a v případě narušení bezpečnosti v zabezpečeném domě se zobrazí červeně přes celá garážová vrata – začne problikávat. Snímač pohybu lze využít také k rozsvícení světla v dolním patře. Pokud bude senzor indikovat pohyb, rozsvítí se světlo na 10 sekund. Po 10 sekundách klidu se zhasne. Pro přepínání funkce snímače pohybu ze zabezpečovacího

systemu na spínač světla slouží tlačítko s textem „*Motion Sensor as Light Switch*“ v pravém horním rohu.

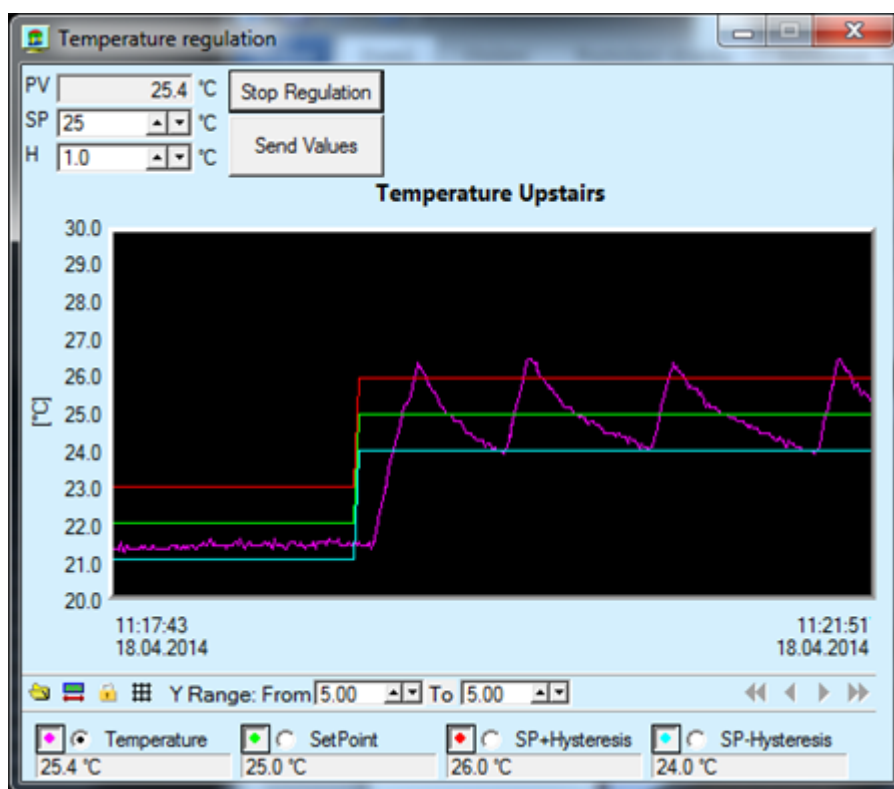


Obrázek 48 – Vzhled aplikace při aktivaci alarmů

Dalším tlačítkem, umístěným v pravém horním rohu s textem „*Roller Blind Simulation*“ popřípadě „*Gate Simulation*“, lze přepínat simulaci otevírání garážových vrat pomocí stejnosměrného motorku nebo zatahování rolet v okně vpravo nahoře na základě nastavené intenzity osvětlení. Pokud je simulace nastavena na ovládání garážových vrat, lze vidět na obrázku tlačítka s šipkami, kterými se vrata otevírají. Tlačítkem nad šipkami se přepíná manuální a automatický režim. Pokud je ovládání v automatickém režimu, tak po stisknutí tlačítka se brána automaticky sama zavře nebo otevře. V manuálním režimu se pohybuje jen tehdy, pokud uživatel tlačítko drží stisknuté. Po přepnutí na simulaci zatahování rolet tlačítka pro ovládání brány zmizí a objeví se aktuální hodnota intenzity osvětlení a nastavení intenzity osvětlení, při které se rolety stáhnou. Hodnoty jsou ve voltech v rozsahu od 0V do 5V. Pokud hodnota intenzity osvětlení překročí danou mez, rolety se stáhnou a naopak při velké tmě se zatahnou. Jak pohyb rolety, tak i garážových

vrát je animován. Stejně tak i pohyb dvoukřídlé brány. Tlačítkem v levém dolním rohu s nápisem „Gate“ se po stisknutí brána začne zavírat nebo otevírat v závislosti na tom, v jaké poloze se zrovna nachází. Pokud se brána pohybuje a tlačítko je stisknuto ještě jednou, brána se zastaví. Po opětovném stisknutí se vrátí zpět do původní polohy.

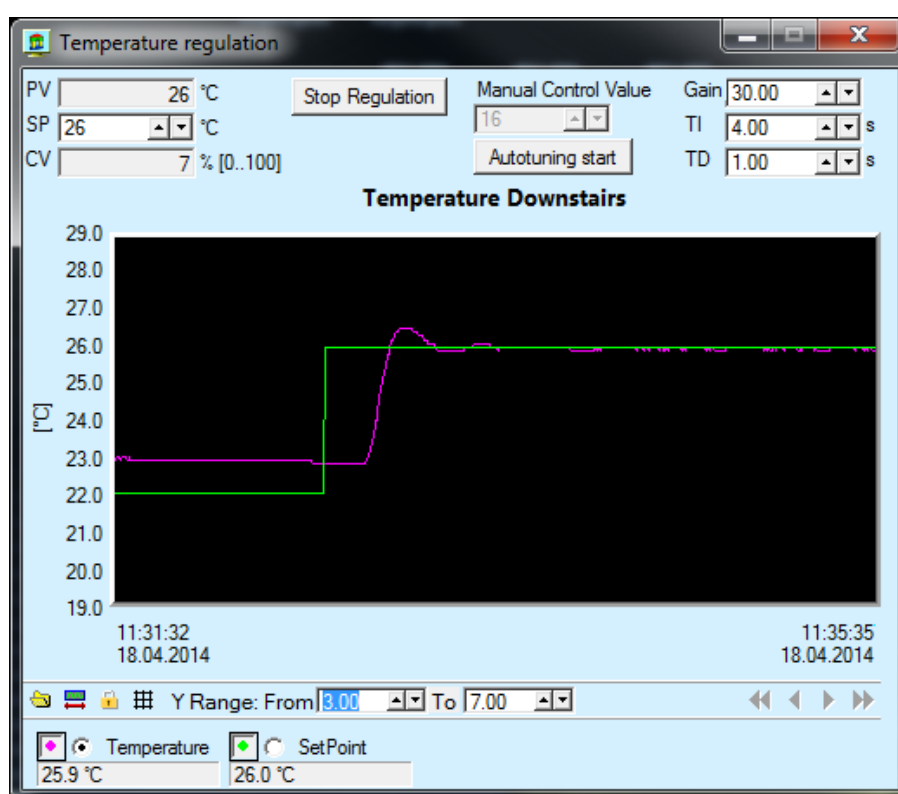
Nakonec jsou na hlavní obrazovce ještě dvě pole zobrazující teplotu ve spodním a v horním patře a dvě tlačítka s texty „Temperature Upstairs“ a „Temperature Downstairs“. Při kliknutí na tlačítko s textem „Temperature Upstairs“ se zobrazí okno sloužící pro regulaci teploty v horním patře. Tato teplota je regulována pomocí jednočipového mikroprocesoru. Je použit algoritmus dvupolohové regulace. Vlevo nahoře je zobrazena aktuální hodnota „PV“, což znamená „Process Value“, požadovaná teplota „SP“ neboli „Set Point“ a hystereze „H“. Při zadání požadované teploty a hystereze se hodnoty musí odeslat do mikroprocesoru PIC pomocí tlačítka „Send Values“. Dále je zde tlačítko, kterým lze regulaci zapnout či vypnout.



Obrázek 49 - Regulace teploty v horním patře pomocí PIC

Při otevření druhého okna sloužícího pro regulaci teploty v dolním patře se zobrazí poněkud pozměněný panel, kde je v horní části zobrazeno více informací, které souvisí s PID regulací teploty pomocí PLC. Jako v předchozím panelu je zde aktuální hodnota teploty „PV“ a požadovaná hodnota teploty „SP“. Dále je zde umístěna indikace hodnoty

akční veličiny „*CV*“ neboli „*Control Variable*“. Ta je zobrazena v procentech. V rámci regulace teploty a nastavení hodnot zesílení („*Gain*“), integrační („*T_I*“) a derivační („*T_D*“) časové konstanty jsou zde dvě možnosti nastavení těchto parametrů. První možností je ruční nastavení těchto parametrů a stisknutím tlačítka „*Start Regulation*“ se regulace spustí. Pokud je regulace vypnutá, lze nastavit hodnotu akční veličiny manuálně, což představuje poměrné rozsvícení žárovky v rozsahu 0-100%. Posledním tlačítkem, u kterého ještě nebyla vysvětlena funkce, je „*Autotuning start*“. Po stisknutí tohoto tlačítka si regulátor najde sám vhodné nastavení parametrů regulace. Po skončení automatického ladění se ihned spustí regulace na základě získaných hodnot.

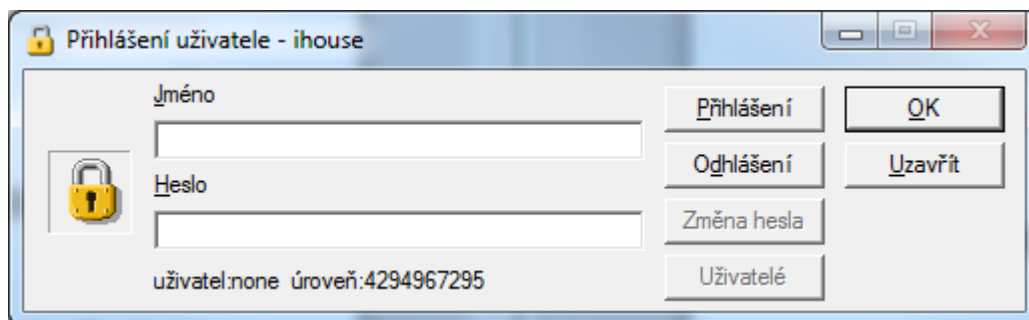


Obrázek 50 - Regulace teploty ve spodním patře pomocí PLC

Trendy, které se zobrazují u obou regulací lze konfigurovat. Muže se nastavit zobrazení mřížky, počet vzorků, který lze zobrazit, rozsah zobrazení y-ové souřadnice od požadované hodnoty. Průběhy se ukládají do samostatných souborů. Po každé hodině je vytvořen soubor nový. Maximální počet souborů k jednomu trendu je 10. Soubory jsou ukládány ve formátu *.dbf. Tento formát lze snadno otevřít v MS Excelu. Samotný přístroj trend po kliknutí na ikonu složky vlevo dole otevře nabídku, kde je možno vidět archivované soubory a nabídku, co s nimi lze provést. Archivní soubor lze tedy otevřít v prohlížeči historických trendů, v InCalcu, což je obdoba MS Excelu a v bBase prohlížeči,

který umožňuje archivovaná data zobrazit ve formě tabulky. Jednotlivé průběhy trendu lze také skrývat pomocí tlačítek s barvou daného průběhu.

V aplikaci je také zaveden systém přístupových práv. Ten se vztahuje na zabezpečení domu. Alarmy se aktivují pouze v případě, že je dům uzamčen po stisku tlačítka s ikonou zámku. Proto, aby uživatel mohl toto tlačítko použít, musí mít dostatečné oprávnění.



Obrázek 51 - Přihlašovací okno

Přihlašovací okno se zobrazí ihned po startu aplikace. Po přihlášení jako správce aplikace lze vytvářet nové uživatele a nastavovat jejich hesla, právo heslo změnit nebo takzvané UID (*User Identifier*). Pomocí čísla UID se určuje typ oprávnění, které uživatel má. Správce má vyhrazené jméno „root“. Pomocí přihlášení se uživatel musí autorizovat také pro přístup k webové aplikaci.

10.2 Webová aplikace

Webová aplikace byla vytvořena pomocí průvodce, kterým lze na základě vytvořeného panelu v Control Web 6 vygenerovat aplikaci stejného vzhledu přístupnou přes webový prohlížeč, a to vše za pomoci komponenty s názvem „*httpd*“. Pro přístup je nutné být v lokální síti a zadat IP adresu počítače, na kterém http server běží. Po dokončení a vygenerování aplikace ale vývoj nekončí, jelikož vygenerovaná webová stránka je optimalizovaná pro prohlížeče na PC. Proto bylo nutno webovou stránku upravit tak, aby byla snadno čitelná i na malých mobilních zařízeních a tabletech.

Aby byl tedy web čitelný, musel se přidat „*meta*“ tag s názvem „*viewport*“, který informuje prohlížeč o tom, že stránka je optimalizovaná pro mobilní zařízení.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=0.9, user-scalable=1">
```


Nastavení atributu „*content*“ slouží pro zadávání parametrů pro změnu vlastností webu v mobilních prohlížečích. Například použitím „*width=device-width*“ se nastaví šířka podle mobilního zařízení.

Dále musel být upraven html kód a to především tak, aby hlavní blok neměl absolutní, ale relativní polohu vůči stránce prohlížeče. Byl také vytvořen nový blok s názvem „*wrapper*“: <DIV ID="wrapper"> a také soubor kaskádových stylů „*style.css*“.

```
@media screen and (max-width: 1920px) {  
  #wrapper {  
    zoom: 1.3;  
  }  
}  
@media screen and (max-width: 550px) {  
  #wrapper {  
    zoom: 1;  
  }  
}
```

Obrázek 52 - Příkazy CSS pro responsivní web

Ten se stará o to, že při zobrazení stránky na mobilních telefonech či tabletech do šířky zařízení 550px se web zobrazí v měřítku 1:1. Při Vyšších rozlišeních je web zvětšen v poměru 1,3:1. Je využíváno takzvaného responsivního webu.

Intelligent House Model Overview

Activate Alarm System
 Motion Detector
 Main Door Window

Temperature Upstairs
 PV °C
 SP °C
 H °C

Temperature Downstairs
 PV °C Gain
 SP °C TI s
 TD s

Lights
 Bedroom Living Room
 Bathroom Kitchen

Garage Gate **Gate**
 Status: Status: Closed

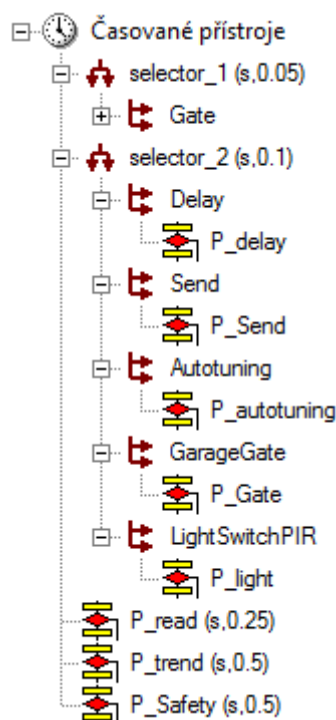
Configuration of Simulation

Obrázek 53 - Vzhled webové aplikace

Vzhled webové aplikace je uživatelsky velmi jednoduchý. Je to jakýsi přehled všech funkcí modelu inteligentního domu. Lze tedy ovládat zabezpečení domu, světla, teplotu nebo garážová vrata a bránu. Aby bylo demonstrováno, že nemusí být přístupny všechny funkce, tak skrze webovou aplikaci nelze nastavovat intenzitu osvětlení, při které se zatáhnou rolety. Pokud je tato simulace aktivní, po kliknutí na tlačítko ovládající garážová vrata se ihned přepne do simulace garážových vrat.

10.3 Časování a popis programu

Aplikace běží v reálném čase. Proto zde byly vytvořeny dva časovače typu „Selector“, které podle podmínek spouští jednotlivé programy.



Obrázek 54 - Časované přístroje

Jsou zde vytvořeny programy *P_delay*, který realizuje sekundové zpoždění a program *P_Send*, který odesílá na výstupní kanál řetězec obsahující žádanou veličinu a hysterezi. Na výstupní kanál je odeslána přepočtená hodnota ze stupňů Celsia na hodnotu odpovídající rozsahu AD převodníku. Princip komunikace již byl popsán v předchozích kapitolách. V tomto prostředí se také provádí přepočet z hodnot A/D převodníku 0-1023 čtené jednočipovým mikroprocesorem na °C. Maximální změřená teplota je tedy 80°C, což odpovídá hodnotě převodníku 1023. Přepočet na °C je tedy podíl $1023/80$, což dá hodnotu 12,7875. Tato konstanta byla upravena tak, aby se teplota v obou patrech o malou hodnotu lišila a aby byly vidět rozdíly. Proto byla použita hodnota 14.

Dále jsou zde programy, které realizují animaci brány a garážových vrat, automatické ladění PID regulátoru a spínání světla na základě pohybového senzoru. Ty jsou taktéž umístěny s časovači typu „*selector*“ a jsou tedy časovány jen za určitých podmínek.

Absolutně jsou časovány pouze tři přístroje typu „*program*“. Program *P_read* čte každou čtvrt sekundu kanály využívané pro OPC komunikaci s PLC a aktivuje přístroje, které jsou na tyto kanály vázány. Dále slouží tento program k čtení a aktivaci alarmů v zabezpečeném domě a k aktivaci motorku zatahujícího či odtahujícího rolety v závislosti na osvětlení. Program *P_trend* každou půl vteřinu aktivuje trendy zobrazující průběh teplot.

11 Závěr

Cílem práce bylo navržení a realizace modelu inteligentního domu, který je řízen pomocí PLC Simatic 314C-2DP a jednočipového mikroprocesoru PIC16F873A. Byly zde popsány programovatelné automaty firmy Siemens. S tím souvisí seznámení se s vývojovým prostředím Step7 pro vývoj aplikací. Dále zde jsou objasněny základní možnosti pro poskytování dat, a to AS-interface a Profibus, protože se v praxi nejčastěji využívají. Jelikož byl jako druhý řídicí systém použit jednočipový mikroprocesor řady PIC, byly charakterizovány základní možnosti těchto procesorů a způsob jejich programování pomocí MikroC. Oba tyto řídicí systémy komunikují se systémem Control Web 6, který slouží jak pro vizualizaci, tak pro ovládání modelu. Proto byl tento systém také stručně popsán.

V následující části byl vytvořen návrh modelu inteligentního domu, který je pomocí vybraného PLC a PIC řízen. V návrhu se objevují základní úlohy inteligentní domácnosti, jako jsou regulace teploty, ovládání rolet, zabezpečovací systém atd. Po vytvoření všech potřebných desek plošných spojů byly vytvořeny dva panely, na které byly umístěny jak všechny desky plošných spojů, tak jednotlivé komponenty simulující funkce inteligentního domu. Na prvním panelu je většina plošných spojů a dále komponenty realizující regulaci teploty, ovládání osvětlení, snímač osvětlení, pohybový senzor a sirénka. Druhý panel obsahuje dvoukřídlovou bránu a roletku, která simuluje buď přímo roletu, nebo ovládání garážových vrat. Dále jsou zde senzory na dveřích a oknu. Celkem bylo vytvořeno 19 desek plošných spojů, jejichž funkce byla v této práci podrobně vysvětlena.

Dalším krokem byl systémový návrh z hlediska programových modulů. V systému Control Web 6 byla vytvořena aplikace, která komunikuje s PIC pomocí sériové komunikace a s PLC pomocí OPC komunikace. Funkce této vizualizace není jen monitorovací, ale také ovládací. V práci byl popsán postup při programování PLC i PIC. Jednočipový mikroprocesor se stará o dvoupolohovou regulaci teploty, snímání hodnoty aktuálního osvětlení a ovládání stejnosměrného motorku. O všechny ostatní signály, například pro PID regulaci nebo ovládání osvětlení, se stará PLC.

Na základě aplikace vytvořené v Control Webu 6 byla vygenerována webová aplikace. Její vzhled byl vytvořen tak, aby byl snadno čitelný a ovladatelný i v mobilních zařízeních. Zdrojový HTML kód webové aplikace musel být upraven tak, aby se v těchto zařízeních správně zobrazoval. Proto bylo využito vlastností takzvaného responsivního webu. Při

určité šířce prohlížeče se aplikace zvětšuje. Toho bylo docíleno podmínkou v CSS souboru.

Co se týče porovnání dvou použitých řídicích systémů, tak PLC má mnoho předdefinovaných funkcí a knihoven, které lze snadno využívat, což se uplatnilo například u PID regulace a analogových vstupů a výstupů. Výhodou je také programování v různých jazycích. Oproti tomu u PIC je nutno znát jeho vnitřní strukturu a o veškeré algoritmy se musí postarat programátor sám. Lze zde taktéž využít knihovny funkcí, ale jen v omezené míře například pro čtení analogové hodnoty nebo pro sériovou komunikaci. Jeho velkou výhodou je oproti PLC cena.

Posledním bodem této práce je navrhnout směr dalšího řešení. Dalším rozšířením by mohlo být využití ZigBee modulů pro bezdrátovou komunikaci mezi jednočipovými mikroprocesory PIC. Jeden mikroprocesor by tedy v případě regulace teploty četl hodnotu ze snímače, posílal ji skrze modul ZigBee nadřazenému mikroprocesoru, který by zpět posílal hodnotu žádané veličiny popřípadě další parametry regulace. Tímto způsobem lze také model rozšířit o další aplikace. Dále by mohla být použita kamera, která by byla součástí zabezpečovacího systému, nebo GSM modul, který by v případě narušení bezpečnosti poslal SMS a prozvonil na zadané číslo.

12 Summary

The aim of this project was to design intelligent family house model which is controlled by PLC Simatic 314C-2DP and microcontroller PIC16F873A. There is description of Siemens PLCs and their developing environment Step7 and PIC microcontrollers and MikroC compiler. Both of control systems communicate with Control Web 6 serving for visualization and for controlling the model.

The design of model contains basic operations of intelligent houses like temperature regulation, control of roller blinds, security system and so on. First, all 19 PCBs have been designed and made and they have been placed on two panels. On these panels are placed all other components simulating functions of intelligent house model. On the first panel are placed almost all of them plus some components implementing temperature regulation with help of PIC and PLC, control of lights, lighting sensor, motion detector and siren. The second panel contains two wings gate, roller blind simulating window blind or garage gate, magnetic sensors on window and door.

Second phase of this work was to design program modules. The application in SCADA system Control Web 6 has been created. Control Web 6 communicates with PLC via OPC communication protocol and with PIC with help of serial communication. This application is not only for viewing but also for controlling. There is a description of PLC and PIC programming too. Microcontroller PIC takes care of on-off temperature control, actual lighting reading and DC motor controlling. PLC takes care of all other signals for example for PID temperature control.

Web application was generated based on panel designed in Control Web 6. Its look is designed so that it is easily controllable and readable from mobile phones or tablets through web browsers. HTML source code had to be modified by reason of right view in these devices. The responsive web properties were used because of that. Size of the web page depends on web browser's width. This was done with help of conditions in CSS file.

Now, it will be compared both control systems. PLC has a lot of predefined functions and libraries which can be easily used. This feature helps for example with PID regulation programming or analog values scaling. Another advantage is programming in different languages. In comparison with microcontroller, PIC programmer has to know inner structure and he has to take care of all functions himself. On the other hand there are

libraries that help for example with serial communication or analog value reading configurations. The biggest advantage compared to PLC is lower price.

At the end, there is proposal of some future possibilities. ZigBee modules can be used for wireless communication between microcontrollers so that intelligent house model would be easily extensible with new applications. Another possibility is to use camera that could be part of security system or GSM module which would send SMS on mobile phone in case of an alarm.

13 Seznam použité literatury

ABB 2011: *Návrhový a instalační manuál systému Ego-R*. [online]. [cit. 2013-12-29]. Dostupné z: <http://www117.abb.com/document.asp?thema=8929>

ANALOG DEVICES 2005: *Low Voltage Temperature Sensors TMP35/TMP36/TMP37*. Datasheet. [online]. [cit. 2013-12-29]. Dostupné z: <http://www.gme.cz/img/cache/doc/530/093/tmp36gt9z-datasheet-1.pdf>

ASIX. Presto - obrázky [online]. Dostupné z: http://www.asix.cz/prg_presto_images.htm

AUTOMA 2004. *Průmyslová komunikační síť AS-interface*. Časopis pro automatizační techniku [online]. Praha: FCC Public, č. 4 [cit. 2013-04-19]. ISSN 1210-9592. Dostupné z: http://www.odbornecasopisy.cz/index.php?id_document=32276

BALÁTĚ, J. 2003. *Automatické řízení*. Praha: Nakladatelství BEN, 654 s. ISBN 80-7300-020-2.

FOXON s.r.o. & Jaroslav Blažek. 2012. *Kurz programování PLC SIEMENS SIMATIC S7-300* [online]. [cit. 2013-04-14]. Dostupné z: <http://www.foxon.cz/>

JANEČEK, J. 1993. *Distribuované systémy*. Praha : Vydavatelství ČVUT, 1993.

KLEČKA R. 2004, *Programová podpora e-modelů z oblasti prostředků automatického řízení*. Bakalářská práce. [online]. [cit. 2013-04-23]. Dostupné z: http://www.e-automatizace.cz/ebooks/ridici_systemy_akcni_cleny/

KOZIOREK, J., CHROMČÁK L. 2007. *Logické systémy řízení a programovatelné automaty*. Ostrava: Vysoká škola báňská - Technická univerzita, 1 DVD-R. ISBN 978-80-248-1490-2.

NACHTIGAL CH. L. *Instrumentation and Control - Fundamentals and Applications*. New York : John Wiley & Sons, Inc. 1993.

MATZ, V. 2010. *Systémy používané v "inteligentních" budovách - přehled komunikačních protokolů*. [online]. [cit. 2014-01-22] Dostupné z: <http://vytapani.tzb-info.cz/mereni-a-regulace/6879-systemy-pouzivane-v-inteligentnich-budovach-prehled-komunikacnich-protokolu>

MICROCHIP TECHNOLOGY INC. 2012, *PIC16F87XA: Datasheet*. [online]. [cit. 2013-12-29]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/39582C.pdf>

MIKROLEKTRONIKA. *MikroC: User's manual* [online]. 2003[cit. 2014-01-25]. Dostupné z: http://www.mikroe.com/pdf/mikroc/mikroc_manual.pdf

MORAVSKÉ PŘÍSTROJE a.s. 2005. *Co je Control Web 5?*. [online]. [cit. 2014-01-26]. Dostupný z www: <http://www.mii.cz/download/doc/cw5cz.pdf>

MORAVSKÉ PŘÍSTROJE a.s. 2010A. *Control Web*. [online]. [cit. 2014-01-26]. Dostupný z www: <http://www.mii.cz/art?id=179&cat=146&lang=405>

MORAVSKÉ PŘÍSTROJE a.s. 2010B. *Manuál Control Web 6*. [online]. Dostupný z www: <http://www.mii.cz/>

PAWLENKA, M. *Řízení, monitorování a vzdálená správa laboratorního modelu: bakalářská práce*. Ostrava: VŠB - technická univerzita Ostrava, Fakulta strojní, Katedra automatizační techniky a řízení, 2012, 51 s. Vedoucí práce: Škuta, J.

SAMIEC T. 2010. *Porovnání komunikačních sítí a metod programování u PLC Simatic*. Bakalářská práce. FEKT VUT v Brně

SIEMENS. 2004. *SIMATIC S7-400*. [online]. [cit. 2013-03-30]. Dostupné z: <http://www.siemens.com/entry/cz/cz/#>

SIEMENS: SIMATIC 2004. *Programming with STEP 7 Lite V3.0 Manual* [online]. 2004 [cit. 2013-04-18]. Dostupné z: <http://www.automation.siemens.com>

SIEMENS 2009. *SIMATIC STEP 7 Professional* [online]. [cit. 2013-04-18]. Dostupné z: <http://www.automation.siemens.com>

SIEMENS. 2012. *Automatizační systémy - Industry Automation & Drive Technologies - Siemens* [online]. [cit. 2013-03-23]. Dostupné z: <http://www.siemens.com/entry/cz/cz/#>

SIEMENS 2013. *Industrial automation systems SIMATIC - Automation Technology - Siemens*: [online]. [cit. 2013-03-23]. Dostupné z: <http://www.automation.siemens.com>

ŠKUTA, J. - MAREK, H. 2006. *Elektronické učební texty pro jednočipové procesory řady PIC* [online] [cit. 2014-01-05] Dostupné z: <http://352lab.vsb.cz/ServerFinalVer/Marek/pic-html/index.htm>

ŠPRINGL, V. 2011. *Mikrokontroléry PIC* [online] [cit. 2014-01-05] Dostupné z: <http://mikrokontrolery-pic.cz/>

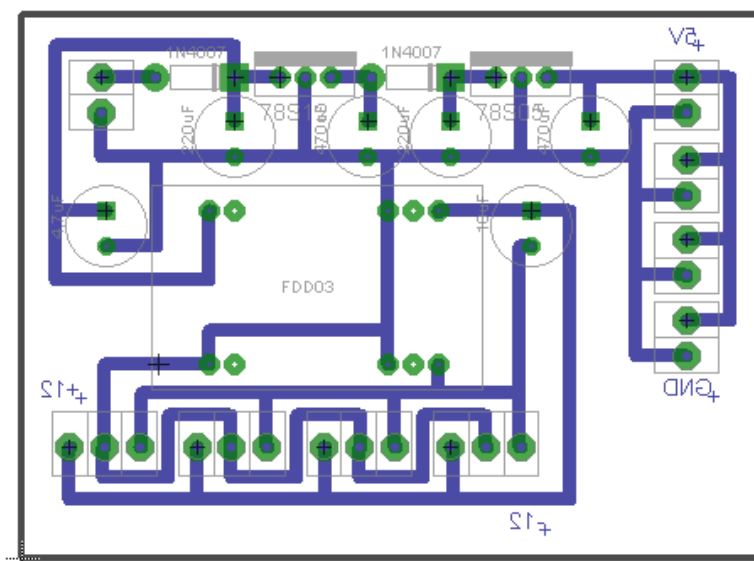
VISHAY 1999. *TCST110. Up to TCST230 Datasheet*. [online], s. 9 [cit. 2014-04-16]. Dostupné z: <http://www.gme.cz/img/cache/doc/521/043/tcst2103-datasheet-1.pdf>

VLACH, J. 1997. *Počítačová rozhraní, přenos dat a řídicí systémy*. Praha, BEN-technická literatura, ISBN 80-85940-17-4.

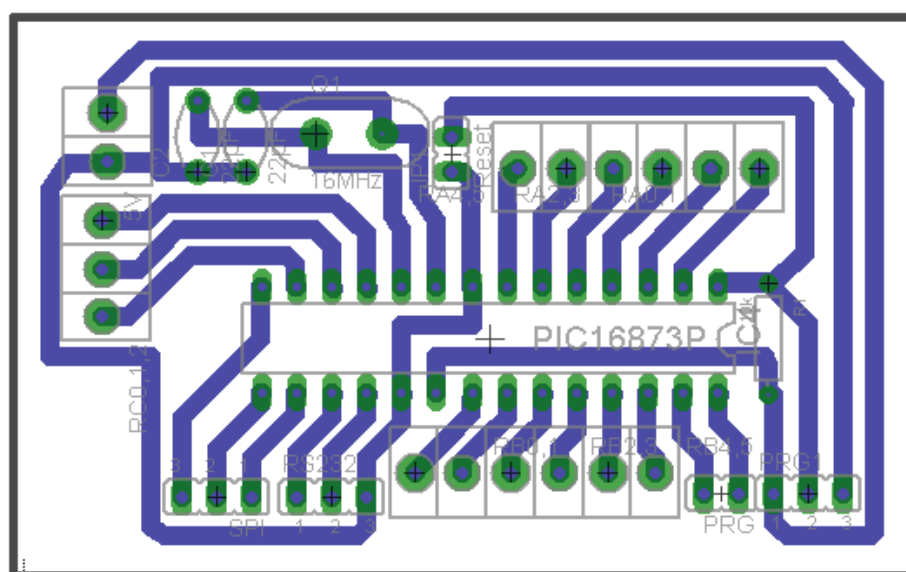
Přílohy

- 1) Návrh desek plošných spojů
- 2) Seznam signálů pro PLC
- 3) Konfigurace ADC modulu pro čtení analogových hodnot z PIC
- 4) Zapojení konektorů mezi panely a k PLC
- 5) STOČ 2014 - Diplom za 2. místo
- 6) Diplom za účast na 51. konferenci studentských prací v Krakově

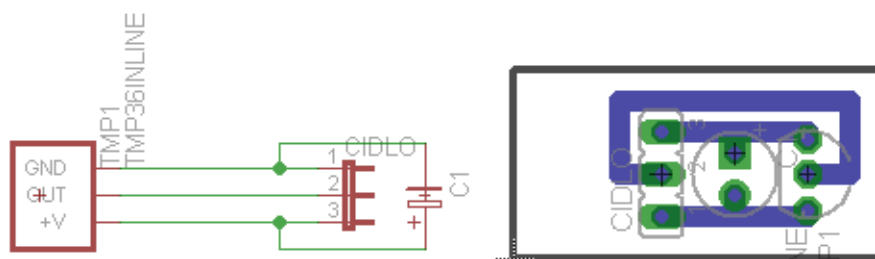
Příloha 1 - Návrh desek plošných spojů



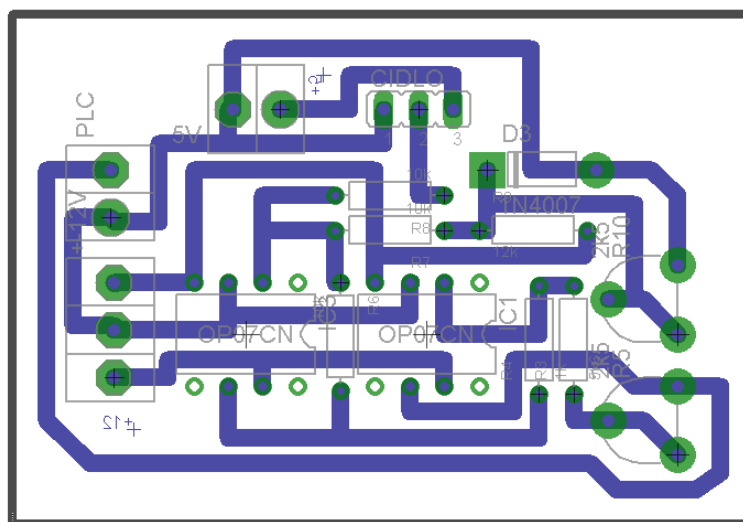
Obrázek 1 - Navržený plošný spoj výkonového modulu



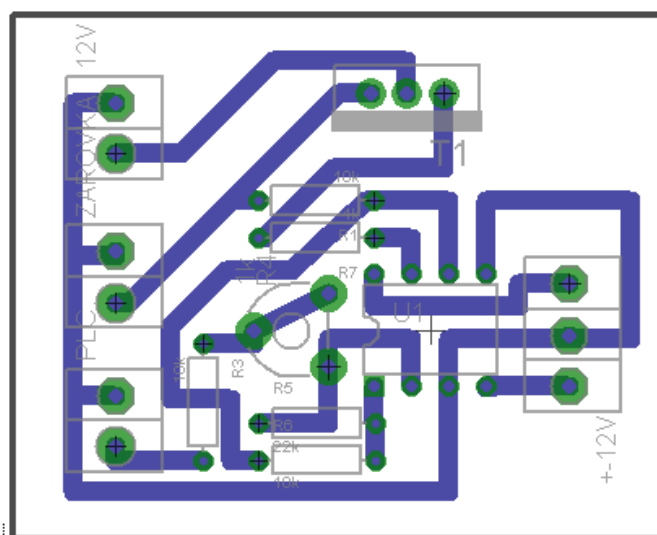
Obrázek 2 - Navržený plošný spoj pro PIC16F873A



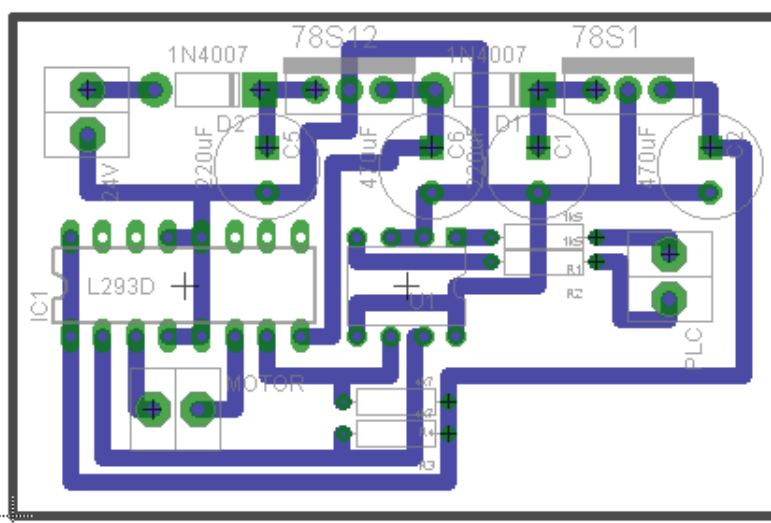
Obrázek 3 - Navržený plošný spoj teplotního snímače TMP36GT9Z



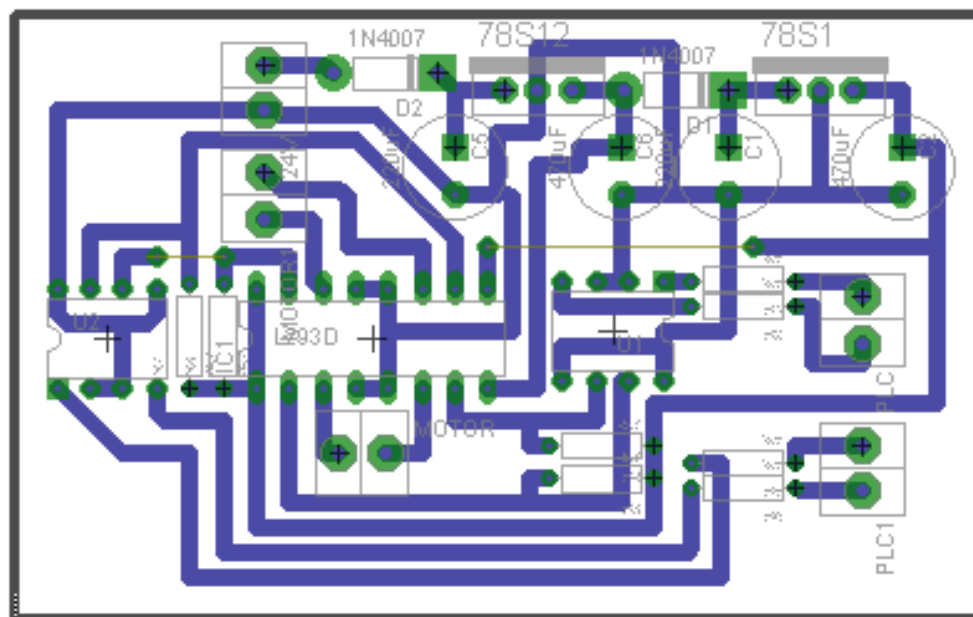
Obrázek 4 - Navržený plošný spoj pro úpravu signálu ze snímače pro PIC



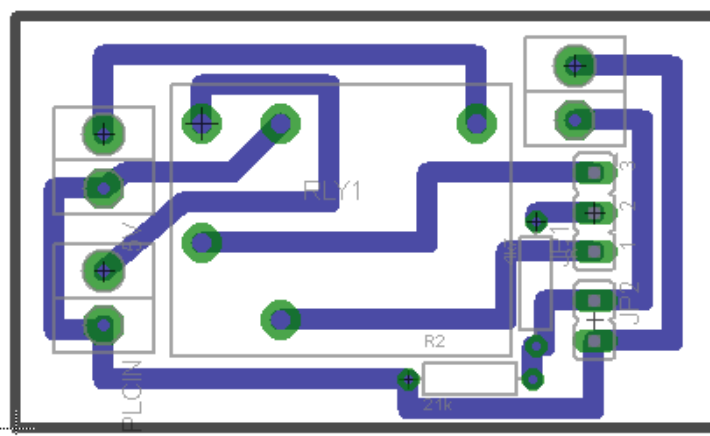
Obrázek 5 - Navržený plošný spoj pro ovládání žárovky z PIC



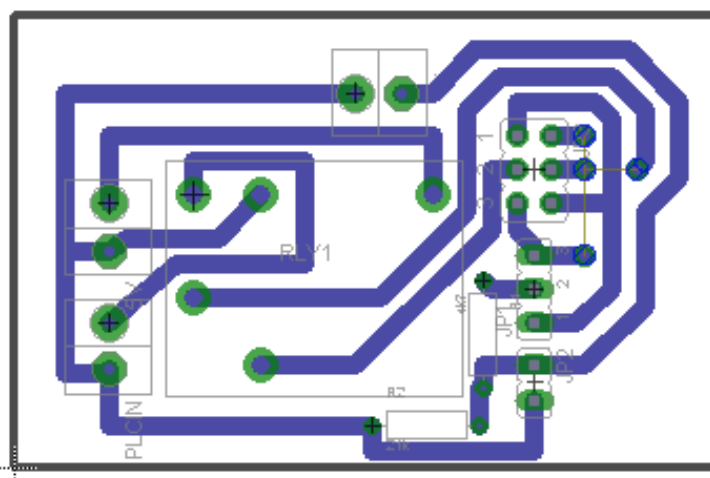
Obrázek 6 - Navržený plošný spoj pro ovládání DC motorku – využit 1 H-můstek



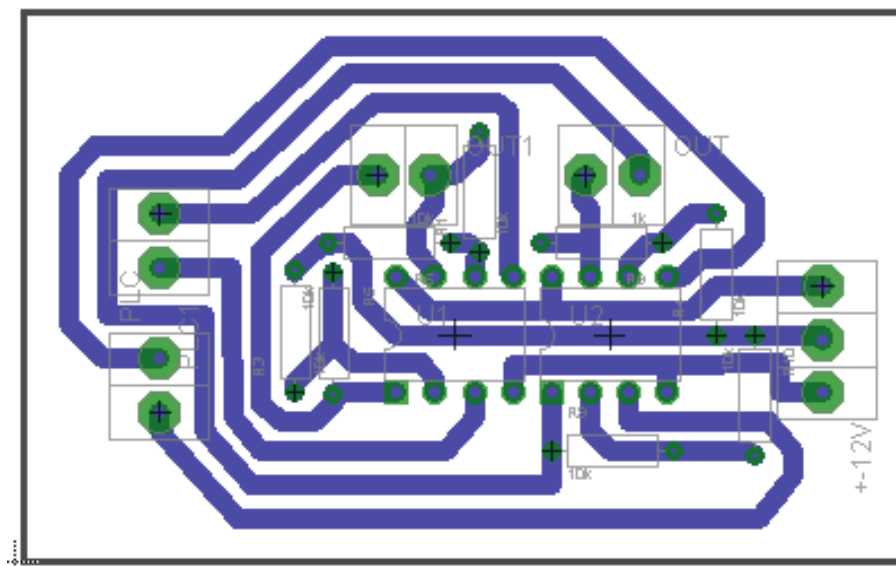
Obrázek 7 - Navržený plošný spoj pro ovládání DC motorku – využity oba H-můstky



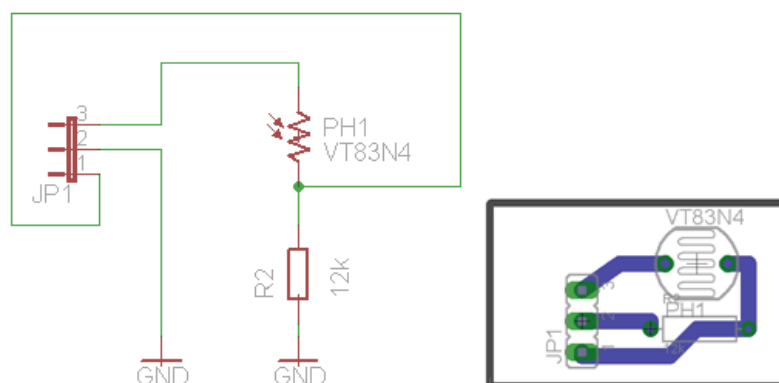
Obrázek 8 - Navržený plošný spoj pro ovládání osvětlení 1 tlačítkem a 1 relé



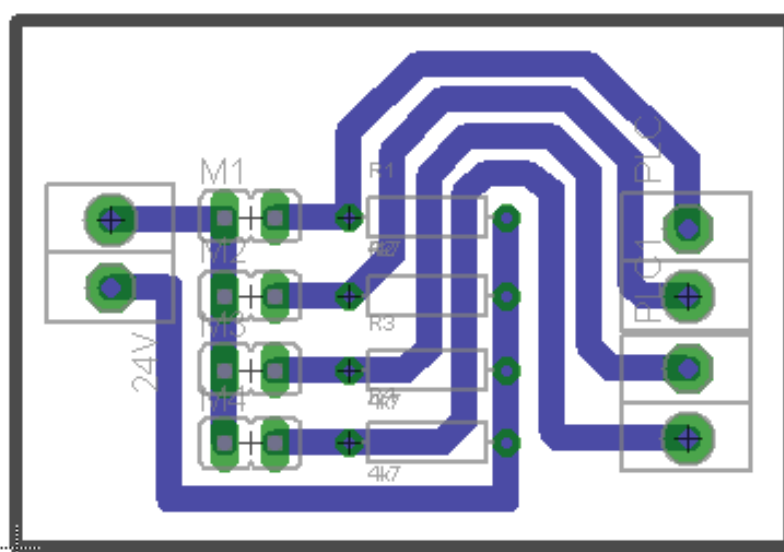
Obrázek 9 - Navržený plošný spoj pro ovládání osvětlení 2 tlačítky a 1 relé



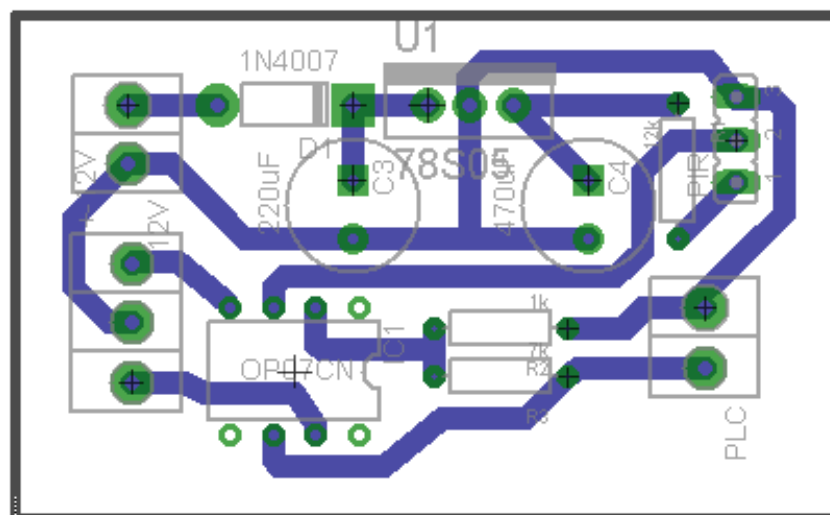
Obrázek 10 - Navržený plošný spoj pro zesílení signálu z LED diod



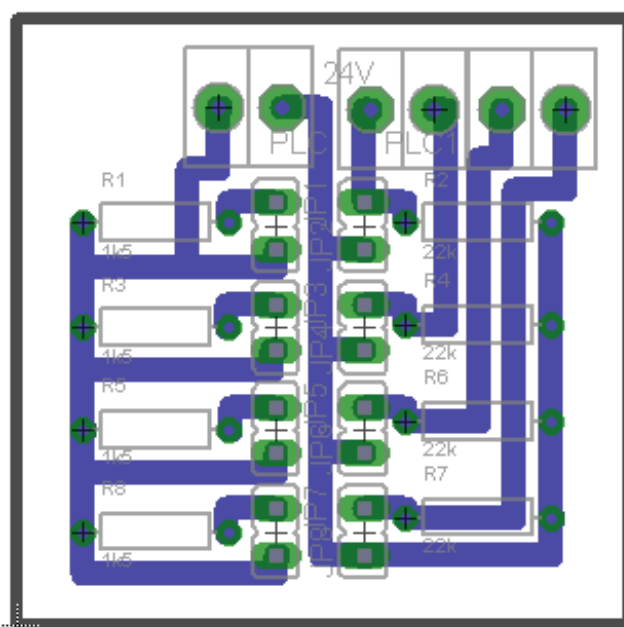
Obrázek 11 - Navržený plošný spoj pro snímač osvětlení



Obrázek 12 - Navržený plošný spoj pro připojení magnetických spínačů



Obrázek 13 - Navržený plošný spoj modulu upravující signál z PIR senzoru



Obrázek 14 - Navržený plošný spoj modulu světelných závor

Příloha 2 – Seznam signálů PLC

Seznam signálů je také v příloženém souboru „IO list.xlsx“.

Tabulka 1 - Seznam signálů pro PLC

Tagname	Address	Data type	Comment
Light1	I 0.0	BOOL	Light 1
Light2	I 0.1	BOOL	Light 2
Light3	I 0.2	BOOL	Light 3
Light4	I 0.3	BOOL	Light 4
PIR	I 0.4	BOOL	PIR - motion sensor indication
MagSensor1	I 0.5	BOOL	Magnetic Sensor 1 - Door
MagSensor2	I 0.6	BOOL	Magnetic Sensor 2 - Window
MagSensor3	I 0.7	BOOL	reserve
MagSensor4	I 1.0	BOOL	reserve
Zav_Mleft_right	I 1.1	BOOL	Left gate closed
Zav_Mright_left	I 1.2	BOOL	Right gate closed
Zav_Mright_right	I 1.3	BOOL	Right gate opened
Zav_Mleft_left	I 1.4	BOOL	Left gate opened
Ain	PIW 252	INT	Read temperature
AOut	PQW 252	INT	Control bulb
CtrlLight1	Q 0.0	BOOL	Light 1 switch
CtrlLight2	Q 0.1	BOOL	Light 2 switch
CtrlLight3	Q 0.2	BOOL	Light 3 switch
CtrlLight4	Q 0.3	BOOL	Light 4 switch
Siren	Q 0.4	BOOL	Siren for alarm
Mleft_open	Q 0.5	BOOL	Left motor open command
Mleft_close	Q 0.6	BOOL	Left motor close command
Mright_open	Q 0.7	BOOL	Right motor open command
Mright_close	Q 1.0	BOOL	Right motor close command

Příloha 3 – Konfigurace ADC modulu pro čtení analogových hodnot z PIC

Nejprve byla vytvořena funkce, která inicializuje registry ADCON0 a ADCON1

```
void ADCInit() {
    ADCON0 = 0b00000101;
    ADCON1 = 0b10000000;
}
```

Registr ADCON0:

ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7							bit 0

Obrázek 55 - Jednotlivé bity ADCON0 registru, viz [MICROCHIP TECHNOLOGY INC. 2012]

Následuje popis jednotlivých bitů tohoto registru.

Tabulka 4 - Význam bitů ADCON0 registru, viz [MICROCHIP TECHNOLOGY INC. 2012]

Bity	Funkce
Bit 0	1 = AD převodník je zapnut 0 = AD převodník je vypnut
Bit 1	Neimplementován
Bit 2	GO/DONE: status bit AD převodu 1 = AD převod právě probíhá (po skončení se automaticky nastaví na 0) 0 = AD převod neprobíhá
Bit 3-5	Výběr analogového kanálu 000 = Kanál 0 (AN0) 001 = Kanál 1 (AN1) 010 = Kanál 2 (AN2) 011 = Kanál 3 (AN3) 100 = Kanál 4 (AN4) 101 = Kanál 5 (AN5) 110 = Kanál 6 (AN6) 111 = Kanál 7 (AN7)
Bit 6-7	Nastavení hodinových impulsů

Registr ADCON1:

ADFM	ADCS2	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

Obrázek 56 - Jednotlivé bity ADCON1 registru, viz [MICROCHIP TECHNOLOGY INC. 2012]

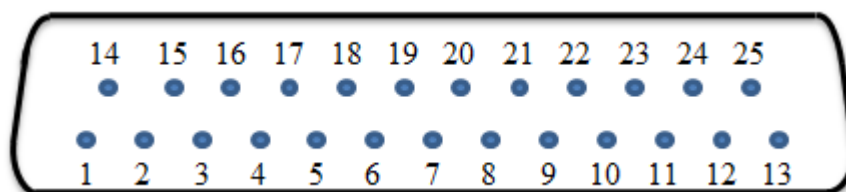
Nyní budou opět popsány jednotlivé bity ADCON1 registru

Tabulka 5 - Význam bitů ADCON1 registru, viz [MICROCHIP TECHNOLOGY INC. 2012]

Bity	Funkce
Bit 0-3	Nastavení jednotlivých analogových pinů mikroprocesoru
Bit 4-5	Neimplementovány
Bit 6	Nastavení hodinových impulsů
Bit 7	Nastavení zarovnání výstupních bajtů 1 = Zarovnání vpravo – 6 nejvíce významových bitů ADRESH jsou 0 0 = Zarovnání vlevo – 6 nejméně významových bitů ADRESL jsou 0

Příloha 4 - Zapojení konektorů mezi panely a k PLC

Schéma zapojení konektoru:



Popis pinů konektoru z panelu 1 do PLC:

- 1 – 4: Indikace osvětlení
- 5: Snímač teploty
- 6: Snímač pohybu PIR
- 7,8,20,21: Magnetické dvevní snímače
- 9,22: Motorek ovládající bránu 1
- 10,23: Motorek ovládající bránu 2
- 11,12,24,25: Světelné závory

Popis pinů konektoru mezi panely 1 a 2:


- 4,10: Motor simulující rolety nebo vrata
- 5,6,17,18: Světelné závory
- 7,19: Motorek ovládající bránu 1
- 8,20: Motorek ovládající bránu 2
- 9,21: Magnetický senzor 4
- 10,22: Magnetický senzor 3
- 11,23: Magnetický senzor 2
- 12,24: Magnetický senzor 1
- 13: Zem
- 25: +24V

Příloha 5 - STOČ 2014 - Diplom za 2. místo



Miroslav Pawlenka


Doc. Dr. Ing. Oldřich Kodym
garant STOČ HGF


Doc. Ing. Renata Wagnerová, Ph.D.
garantka STOČ FS



Příloha 6 – Diplom za účast na 51. konferenci studentských prací v Krakově



AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA
W KRAKOWIE

DYPLOM UCZESTNICTWA

W 51 KONFERENCJI
STUDENCKICH KÓŁ NAUKOWYCH
PIONU HUTNICZEGO

Sekcja Automatyki i Robotyki

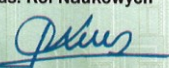
Miroslav PAWLENKA

Temat referatu:

USE OF PLC AND PIC FOR REALIZATION OF INTELLIGENT FAMILY HOUSE


Przewodniczący Jury

Pełnomocnik Rektora
ds. Kół Naukowych


dr inż. Leszek Kurcz

Kraków 2014